

Sample Movie Database Spring Boot Project

Team 05

Project
Future

 Code.Hub

Agenda

1. Project Developers
2. Domain Model
3. Entities & Database
4. Project Structure
5. Crud & Data Backup
6. Reports
7. Demo Time!
8. Future Work

Project Developers

Members

Stylianos
Stamatakis
Crete



Efstathios
Antonakis
Thessaloniki



Marianna
Manou - Kaklamani
Athens



Ioannis
Pagonis
Patra



Spyridon
Massios
Athens



Domain Model

Domains

Person

- FirstName (String)
- lastName (String)
- YOB (Integer)
- country (String)
- isAlive (Boolean)

Film

- title (String)
- description (String)
- releaseYear (Integer)
- language (String)
- duration (Integer)
- rating (Rating)
- **@CollectionTable**
<Set> genre (Genre)
- **@OneToMany**
<Set> peopleCasted (CastedPerson)

TvShow

We noticed that the only difference between films and tv shows are the tv show seasons and episodes, hence TVShow extends Film.

- numberOfSeasons (Integer)
- numberOfEpisodes (Integer)

Classes & Enums

Casted Person

- **@ManyToOne**
person (Person)
- **@ManyToOne**
film (Film)
- role (Role)

Role (E)

- Actor
- Director
- Producer
- Writer
- BoomOperator
- FilmEditor
-

Rating (E)

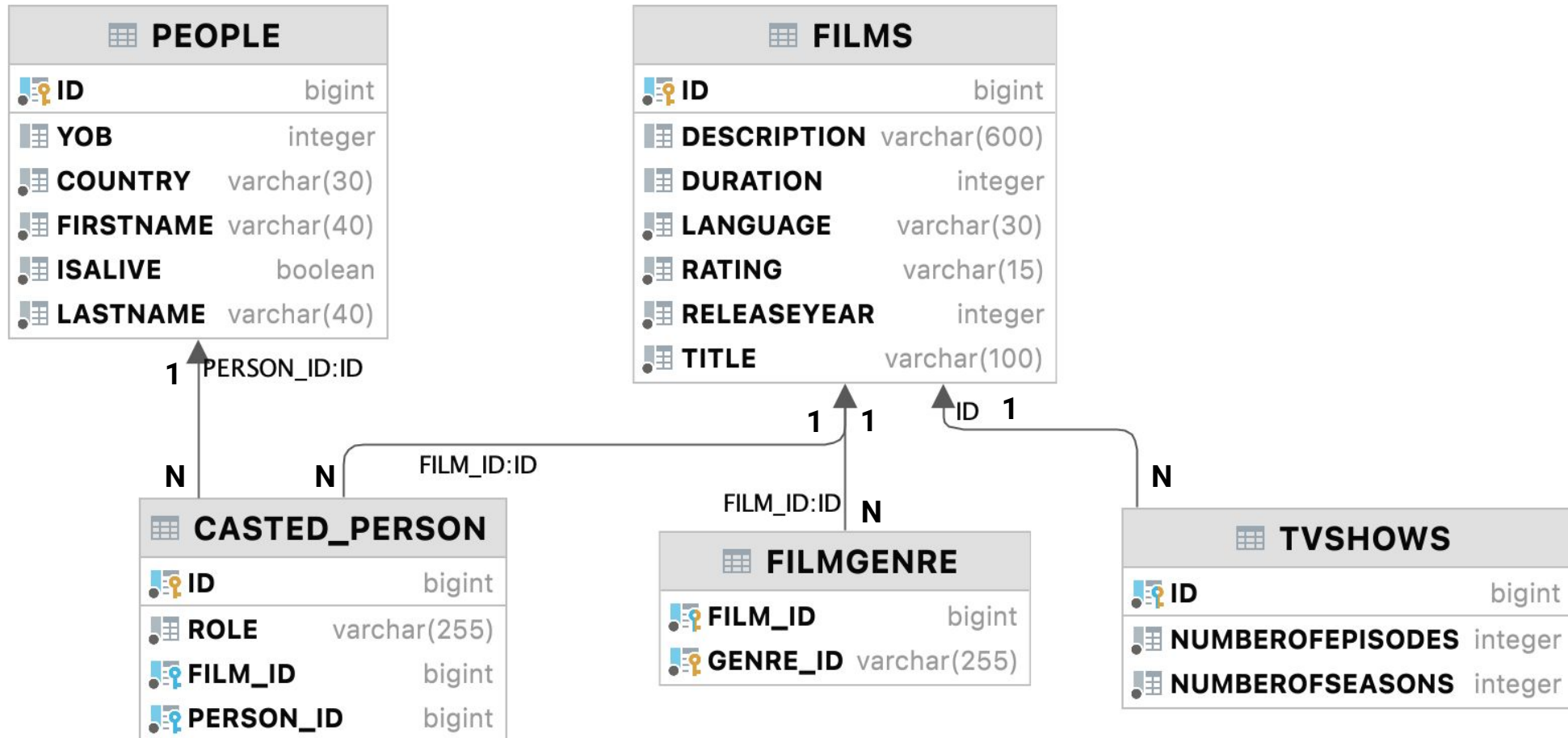
- one
- two
- three
- four
- five

Genre (E)

- Romance
- Drama
-
- Thriller
- Animation

Entities & Database

ER Diagram



Project Structure

Maven Dependencies

- We used two separate pom.xml files one for the “parent” model and one for the “child” module.
- The dependencies placed inside the “parent” file was the one that we wanted to apply to all the modules.
- This way the we have smaller files that include relative information based on the service of the module.

Parent module

```
<spring-boot-configuration-processor>      <log4j-api>
      <spring-boot-starter-logging>
    <lombok>
      <spring-boot-starter-log4j2>      <log4j-core>
      <disruptor>
```

Child module

```
<super-csv>
<spring-boot-starter-web>
<spring-boot-starter-data-jpa>
<h2>
<opencsv>
<spring-boot-starter-data-rest>
<spring-boot-starter-validation>
<assertj-core>
<json-simple>
<freemarker>
<spring-context-support>
```

Folders

bootstrap

- GenerateFilmAndTVshow and GeneratePeople are employed to populate the database with films, tv shows and people with random field values.



For the development testing we created a SQL file called **import.sql**. This way, we can easily test our queries because we know what output to expect.

service

- Based on the domain classes.
- Base Service handles the basic functionality.
- Reports are implemented in FilmServiceImpl.
- Using logging we display informative messages.



Folders

repository

- One repository for each domain class.
- Each repository has queries for its respective domain.
- JPA handles the basic crUd operations.
- JPQL was used for simple queries.
- Native queries were used for more complex ones.
- Report queries take place in FilmRepository.

▼  repository

-  FilmRepository
-  PersonRepository
-  TVShowRepository

controlles

- Each domain class has its own controller.
- We included an abstract controller for basic functionality.
- ExportController handles the CSV files.
- ExceptionHandler handles the exceptions in all controllers.

▼  controller

-  AbstractController
-  CustomizedExceptionHandler
-  ExportController
-  FilmController
-  PersonController
-  ReportsController
-  TVShowController

Coffee Break



Crud & Data Backup

Crud Operations

- Each domain class have their own crUd operations for querying the associated information
- The information is passed and sorted based on the arguments and header values on the request.
- Based on the role and importance of the domain we can add more operations.

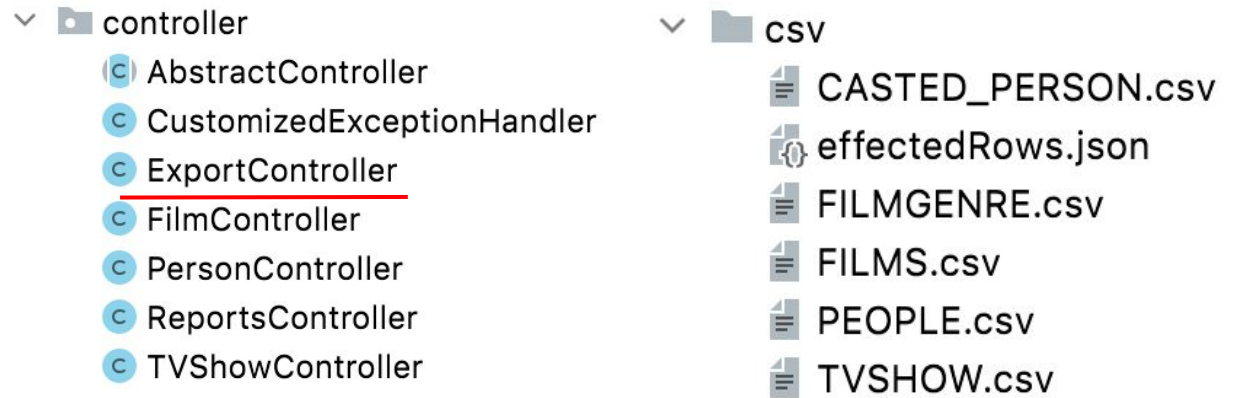
```
✓ Project Future 7 - Spring Boot Project
  > Reports
  > Export CSV
  ✓ crUd Operations
    ✓ TVshow
      GET GET TVshow by title
      GET GET TVshow by episodes
      GET GET TVshow by seasons
    ✓ Person
      GET GET Person by name
      GET GET Person from country
      GET GET Persons Casted Films
      GET GET Person by isAlive
      PUT PUT People
    ✓ Film
      GET GET TVshow by title
      DEL DELETE Film
```


Creating CSV Files

Using a simple REST call the database is exported to CSV files. The created files are stored inside a new directory called “csv”. The naming of the files is based on the associated table name for convenience.

Lastly a JSON file is created, reporting back the number of rows exported per domain class.

***Altering the REST call only a specific table can be exported from the database. (e.g. PEOPLE, FILM etc)**



PEOPLE.csv

	C1	C2	C3	C4	C5	C6
1	Person ID	First Name	Last Name	YOB	Country	Is Alive
2	1	Damiano	Moretti	1960	Italy	false
3	2	John	Brown	1986	United Kingdom	true
4	3	Magnus	Carlsen	1989	Norway	true
5	4	Amelie	Trindeux	1942	France	false

effectedRows.json

```
1 {  
2   "FILMS":10,  
3   "PEOPLE":11,  
4   "TVSHOWS":2,  
5   "CASTED_PERSON":46,  
6   "FILMGENRE":36  
7 }
```

Reports

Report 1

Project Future 7 - Spring Boot Project / Reports / FirstReport

Save

Send

GET {{url}}/reports?num=2

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION		Bulk Edit
<input checked="" type="checkbox"/>	num	2			

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 47 ms Size: 1.53 KB Save Response

Pretty Raw Preview Visualize JSON

```
3   "createdAt": "2022-01-12 04:05:23.729",
4   "data": {
5     "content": [
6       {
7         "id": 8,
8         "title": "Doctor Strange",
9         "description": "A super hero movie",
10        "releaseYear": 2016,
11        "language": "English",
12        "duration": 26,
13        "rating": "five",
14        "genre": [ ...
15      ],
16    ],
17    "peopleCasted": [ ...
18  ]
19  },
20  {
21    "id": 4,
22    "title": "Pirates of the Caribbean, the curse of the black pearl",
23    "description": "A movie about pirates",
24    "releaseYear": 2003,
25    "language": "English",
26    "duration": 25,
27    "rating": "five",
28    "genre": [
29      "Adventure"
```

Report 2

Project Future 7 - Spring Boot Project / Reports / **SecondReport**

GET {{url}}/reports?id=2 Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> id	2			

Body Cookies Headers (5) Test Results Status: 200 OK Time: 39 ms Size: 1.08 KB Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 1,  
7      "title": "Are you alone ?",  
8      "description": "Do not watch this alone",  
9      "releaseYear": 2016,  
10     "language": "English",  
11     "duration": 136,  
12     "rating": "four",  
13     "genre": [ ...  
17   ],  
18     "peopleCasted": null  
19   },  
20   { ...  
35   },  
36   {  
37     "id": 9,  
38     "title": "Space Jump 3",  
39     "description": "Luca and Bugs Bunny ",  
40     "releaseYear": 2026,  
41     "language": "English",  
42     "duration": 113,  
43     "rating": "four",  
44     "genre": [  
45       "Comedy",  
...
```

Project Future 7 - Spring Boot Project / Reports / **SecondReport**

GET {{url}}/reports?id=2 Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> action	secondReport				
<input checked="" type="checkbox"/> ExtendedInfo	true				

Body Cookies Headers (5) Test Results Status: 200 OK Time: 46 ms Size: 5.09 KB Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 1,  
7      "title": "Are you alone ?",  
8      "description": "Do not watch this alone",  
9      "releaseYear": 2016,  
10     "language": "English",  
11     "duration": 136,  
12     "rating": "four",  
13     "genre": [ ...  
17   ],  
18     "peopleCasted": [ ...  
115  ],  
116  },  
117  { ...  
205  },  
206  {  
207    "id": 9,  
208    "title": "Space Jump 3",  
209    "description": "Luca and Bugs Bunny ",  
210    "releaseYear": 2026,  
211    "language": "English",  
212    "duration": 113,  
213    "rating": "four",  
214    "genre": [ ...  
...
```

Report 3

Project Future 7 - Spring Boot Project / Reports / ThirdReport

GET {{url}}/reports?id=1&role=Actor

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	1			
<input checked="" type="checkbox"/>	role	Actor			

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 94 ms Size: 700 B Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 1,  
7      "title": "Are you alone ?",  
8      "description": "Do not watch this alone",  
9      "releaseYear": 2016,  
10     "language": "English",  
11     "duration": 136,  
12     "rating": "four",  
13     "genre": [...]  
14   },  
15   ],  
16   "peopleCasted": null  
17 },  
18 ],  
19 },  
20 },  
21 },  
22 },  
23 },  
24 },  
25 },  
26 },  
27 },  
28 },  
29 },  
30 }
```

Project Future 7 - Spring Boot Project / Reports / ThirdReport

GET {{url}}/reports?id=1&role=Actor

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Headers 6 hidden

	KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	action	thirdReport				
<input checked="" type="checkbox"/>	ExtendedInfo	true				

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 40 ms Size: 2.55 KB Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 1,  
7      "title": "Are you alone ?",  
8      "description": "Do not watch this alone",  
9      "releaseYear": 2016,  
10     "language": "English",  
11     "duration": 136,  
12     "rating": "four",  
13     "genre": [...]  
14   },  
15   ],  
16   "peopleCasted": [...]  
17 },  
18 },  
19 },  
20 },  
21 },  
22 },  
23 },  
24 },  
25 },  
26 },  
27 },  
28 },  
29 },  
30 }
```

Report 4

Project Future 7 - Spring Boot Project / Reports / FourthReport

GET {{url}}/reports?genre=Action

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	genre	Action			

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 56 ms Size: 1.13 KB Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 6,  
7      "title": "Naruto",  
8      "description": "A show about ninjas",  
9      "releaseYear": 1999,  
10     "language": "Japanese",  
11     "duration": 25,  
12     "rating": "four",  
13     "genre": [  
14       "Comedy",  
15       "Fantasy",  
16       "Action",  
17       "Animation"  
18     ],  
19     "peopleCasted": null,  
20     "numberOfEpisodes": 2,  
21     "numberOfSeasons": 14  
22   },  
23   {  
24     "id": 7,  
25     "title": "The Last Samurai",  
26     "description": "A movie about samurais",  
27     "releaseYear": 2001,  
28     "language": "English",
```

Project Future 7 - Spring Boot Project / Reports / FourthReport

GET {{url}}/reports?genre=Action

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	action	fourthReport				
<input checked="" type="checkbox"/>	ExtendedInfo	true				

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 55 ms Size: 3.21 KB Save Response

Pretty Raw Preview Visualize JSON

```
4  "data": [  
5    {  
6      "id": 6,  
7      "title": "Naruto",  
8      "description": "A show about ninjas",  
9      "releaseYear": 1999,  
10     "language": "Japanese",  
11     "duration": 25,  
12     "rating": "four",  
13     "genre": [  
14       "Comedy",  
15       "Fantasy",  
16       "Action",  
17       "Animation"  
18     ],  
19     "peopleCasted": [...  
56   ],  
57   "numberOfEpisodes": 2,  
58   "numberOfSeasons": 14  
59 },  
60 {  
61   "id": 7,  
62   "title": "The Last Samurai",  
63   "description": "A movie about samurais",  
64   "releaseYear": 2001,
```

Report 5

Project Future 7 - Spring Boot Project / Reports / FifthReport

Save

GET

{{url}}/reports

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Headers

6 hidden

	KEY	VALUE	DESCRIPTION		Bulk Edit	Presets
<input checked="" type="checkbox"/>	action	fifthReport				
	Key	Value	Description			

Body

Cookies

Headers (5)

Test Results

Status: 200 OK

Time: 70 ms

Size: 435 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16



17

18

19

```
"transactionId": "98ED6A24-A83E-4DBE-B5E6-592BABEB5D8A",
"createdAt": "2022-01-12 04:22:14.666",
"data": {
  "Psychological": 1,
  "Action": 4,
  "Adventure": 3,
  "Horror": 2,
  "Thriller": 2,
  "Crime": 1,
  "Fantasy": 6,
  "SliceOfLife": 1,
  "Animation": 6,
  "Romance": 2,
  "Comedy": 6,
  "SciFi": 2
},
"apiError": null
```

Report 6 & 7



Project Future 7 - Spring Boot Project / Reports / **SixthReport**  

GET Send



Params Authorization **Headers (7)** Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	action	sixthReport				

Body Cookies Headers (5) Test Results Status: 200 OK Time: 31 ms Size: 1.53 KB Save Response

Pretty Raw Preview Visualize JSON  

```
4  "data": {
5    "2016": {
6      "Psychological": 1,
7      "Action": 1,
8      "Adventure": 1,
9      "Horror": 2,
10     "Thriller": 2,
11     "Crime": 1,
12     "Fantasy": 1,
13     "SliceOfLife": 1,
14     "Animation": 1,
15     "Romance": 2,
16     "Comedy": 1,
17     "SciFi": 1
18   },
19   "2001": { --
32   },
33   "2003": {
34     "Psychological": 1,
35     "Action": 1,
36     "Adventure": 1,
37     "Horror": 2,
38     "Thriller": 2,
39     "Crime": 1,
40     "Fantasy": 1,
41     "SliceOfLife": 1,
42     "Animation": 1,
```



Project Future 7 - Spring Boot Project / Reports / **SeventhReport**  

GET Send

Params **Authorization** Headers (7) Body Pre-request Script Tests Settings Cookies

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	id	1			

Body Cookies Headers (5) Test Results Status: 200 OK Time: 29 ms Size: 1.88 KB Save Response

Pretty Raw Preview Visualize JSON  

```
4  "data": {
5    "Psychological": [
6      {
7        "id": 10,
8        "title": "Peanut Butter",
9        "description": "Best movie no one knows",
10       "releaseYear": 2014,
11       "language": "English",
12       "duration": 129,
13       "rating": "three",
14       "genre": [],
15       "peopleCasted": []
16     }
17   ],
18   "Horror": [ --
41   ],
42   "Thriller": [
43     {
44       "id": 1,
45       "title": "Are you alone ?",
46       "description": "Do not watch this alone",
47       "releaseYear": 2016,
48       "language": "English",
49       "duration": 136,
50       "rating": "four",
51       "genre": [],
52       "peopleCasted": []
```


Demo Time!

Future Work?



Domains

Person

- FirstName (String)
- lastName (String)
- YOB (Integer)
- country (String)
- isAlive (Boolean)

Film

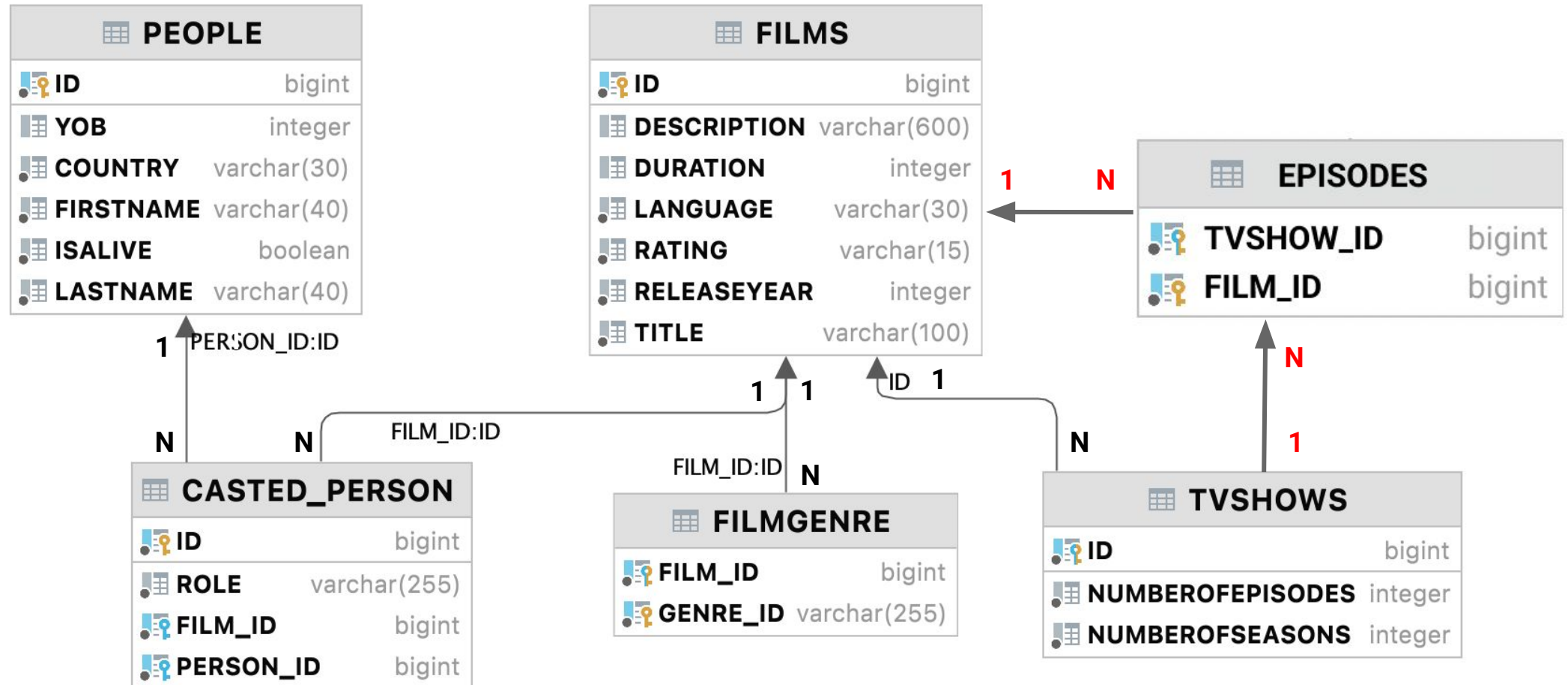
- title (String)
- description (String)
- releaseYear (Integer)
- language (String)
- duration (Integer)
- rating (Rating)
- `@CollectionTable`
• `<Set>` genre (Genre)
- `@OneToMany`
• `<Set>` peopleCasted (CastedPerson)

TvShow

We noticed that the only difference between films and tv shows are the tv show seasons and episodes, hence TVShow extends Film.

- numberOfSeasons (Integer)
- numberOfEpisodes (Integer)
- `@OneToMany`
• `<Set>` episodes (Film)

ER Diagram



Gitlab Repo

Projects github page:

<https://github.com/Cretanian/Sample-Movie-Database-Spring-Boot-Project>

Project Repo:

<https://github.com/Cretanian/Sample-Movie-Database-Spring-Boot-Project.git>

Thank you!



@Questions()