# Movies

1. **Id**
2. Title
3. Director
4. Description
5. **Genre id**
6. Based on book - Boolean?
7. **Book id** – can be null
8. **Rank id**

# Movie actors

1. Id
2. **MovieId**
3. **ActorId**

# Actors

1. **Id**
2. Name
3. Biography
4. //Може да добавим списък с най-известните филми в които играе даден актьор? Тъкмо ще може да си добавим още някоя таблица.

# Genres

1. **Id**
2. Name

# Books

1. **Id**
2. Name
3. Author
4. Category

# Ranks

1. **Id**
2. Name

# Users

1. Id
2. Username
3. Password (hash?)

# User favorite movies

1. Id
2. User id
3. Movie id

# Requirements

## General Requirements

* Use Code First approach
* Use Entity Framework
* Use SQL Server 2016
* At least five tables in the SQL Server database
* Provide all type of relations in the database and use both attributes and the Fluent API (Model builder) for configuration
* Put some of the data of your application in SQLite and PostgreSQL (at least two tables)
* Load some of the data from external files (Excel, XML, JSON, zip, etc.) of your choice – струва ми се че json е най-удачния избор ☺ съдейки по изискванията май и хмл ще е хубаво да вкараме
* The user should be able to manipulate the database through the client (basic CRUD. No need to provide roles for the users)
* Provide some usable user interface for the client (could be the console, WPF, Windows Forms, ASP.NET Web application, etc.)
* Generate PDF reports based on your application – може да използваме iTextSharp  тъкмо и ще покрием 1вото от additional requirements

## Additional Requirements

* For the PDF export use a non-commercial third party framework
* The XML files should be read / written through the standard .NET parsers (of your choice)
* For JSON serializations use a non-commercial library / framework of your choice

## Optional Requirements (Bonus points)

* Unit test the application (think about how and use the techniques from the previous courses in Telerik Academy and the [Design Patterns course](http://telerikacademy.com/Courses/Courses/Details/431))
* You could use Repository pattern or Services of your choice or use pure Entity Framework and DbContext (research the options and choose your preferable way to do it)
* Completely finished and working project is not a requirement but it will be considered a plus