MARMARA
UNIVERSITY

FACULTY OF ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

CSE333-DATABASE SYSTEMS

PROJECT REPORT

STUDENTS

150114048 – Mehmet Cumali DEMİR

150115048 – Muhammed KILIÇ

Submission Date: 15.12.17

INTRODUCTION

The purpose of establishing such a database is to adjust the sound of the society and the cinema to the same frequency. When people decide to watch a movie in the hustle and bustle of everyday life, choosing the movies the time they spend will be longer than watching the movie, because the cinema's language could not address the gathering. The topics and abstracts of movies that we have developed a solution for and the main source of the problem. Movies are a bit difficult to watch in a style that I would like to accumulate in a classical category like "drama, romantic, mystery, surrealist, modern, postmodern or gothic". IMBD is using the typical categories that has a mission like encyclopedia and just aims to collect information for archiving. Therefore, the category classification of films is "not at this stage, should be watched with boy/girl-friend, one-sided watching, brain-burning films, sod head flow, enjoyment, etc." we can introduce ordinary people in society like the magic world of cinema. This database will also have nine asset tables. In addition, the relationships will be established on the "Movie” and “User” tables.

TABLE EXPLANATIONS

1-) Category: There are two qualities. One of these is the category and the name. Storing categories according to their id and assigning one or more categories to a filename.

2-) User: There are seven attributes. To keep the names and personal information of the user ID, name, mailAddress, birthdate, job, age and password users, and to recommend a custom filename according to the information of the person.

3-) Movie: movieID, movieName, categoryID, castID. To create the movie’s originality and movie fiction. Again, originally, encouraging the user to watch the movie is a criticism that can be quickly returned after watching in a promotional article section.

4-) Cast: There are three qualities. directorName, producerName, writerName. Seeing a movie cadre like it very much and seeing it in a collective way to see the director's screenplays.

5-) Award: There are two attributes as awardID and awardName. To see the prizes of your favourite movies or to list the films chosen by the companies that have been recognized as an authority.

6-) Film_Award: There are two attributes. MovieId and AwardID.

7-) Job: There are two attributes as jobID and jobName.

8-) Will_Watch: It is a want to watch list for users. It has movieID and userID attributes.

9-) Watched: It is watched movies list for users. It has movieID and userID attributes.

ATTRIBUTE FEATURES

Indices: User->userMail

Uniques: User->userMail

Identities: Movie-> movieID

Check Constraints: User->userAge

Defaults: User->userJob

Computed Columns: User-> userAge

Triggers: Watched->isExistWill_Watch , Will_Watch-> isExistWatched

*Only necessary attributes are listed.

VIEWS

View Table 1: Statistics for each category number of users who are over 18 and willing to watch.

View Table 2: Movies in a category that has more than two followers and earned two awards.

View Table 3: Other movies seen by a user that have been watched and have same director in cast.

View Table 4: Movies a user watched with less than two awards except who watched movies with more than two awards.

*Can be found in VIEW.sql

TRIGGERS

Trigger 1: Movies on Will_Watch table cannot be on Watched table.

Trigger 2: Movies on Watched table cannot be on Will_Watch table.

*Can be found in TRIGGER_AND_INDEX.sql

STORED PROCEDURES

Procedure 1: Movies have won the same award.

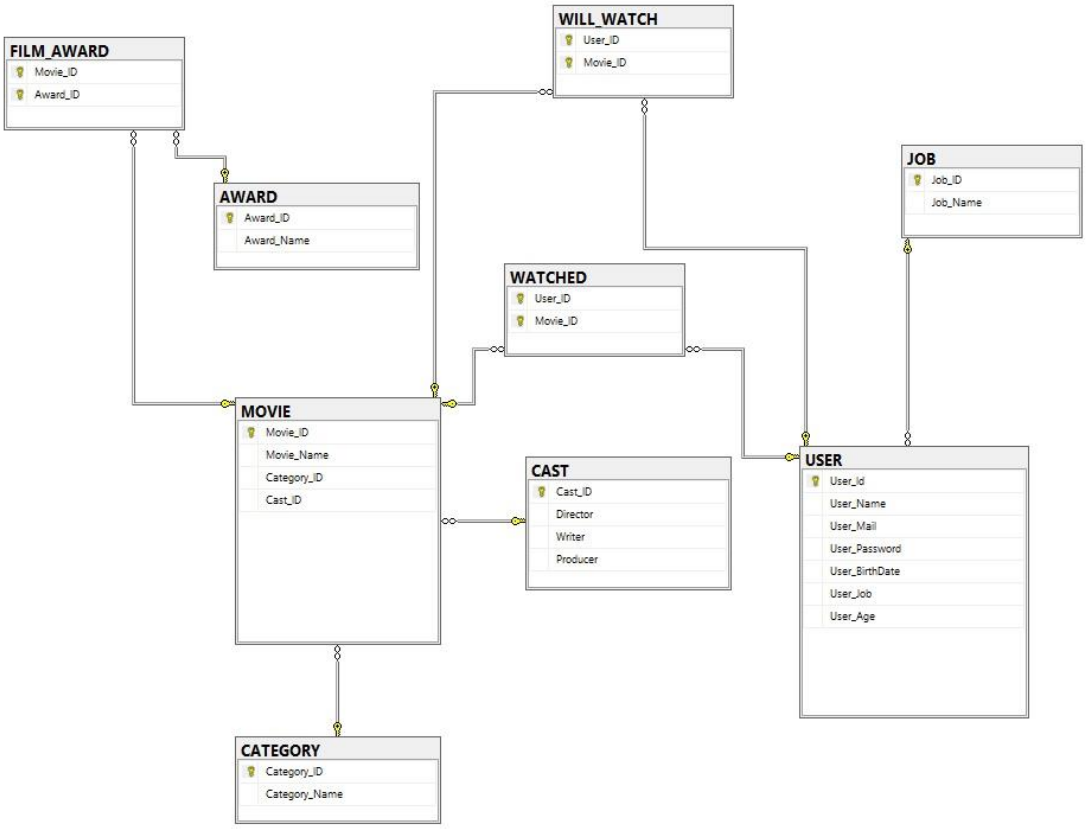
Procedure 2: Add movie and connect it.

Procedure 3: Empty the lists (Will Watch and Watched) when a user deleted.

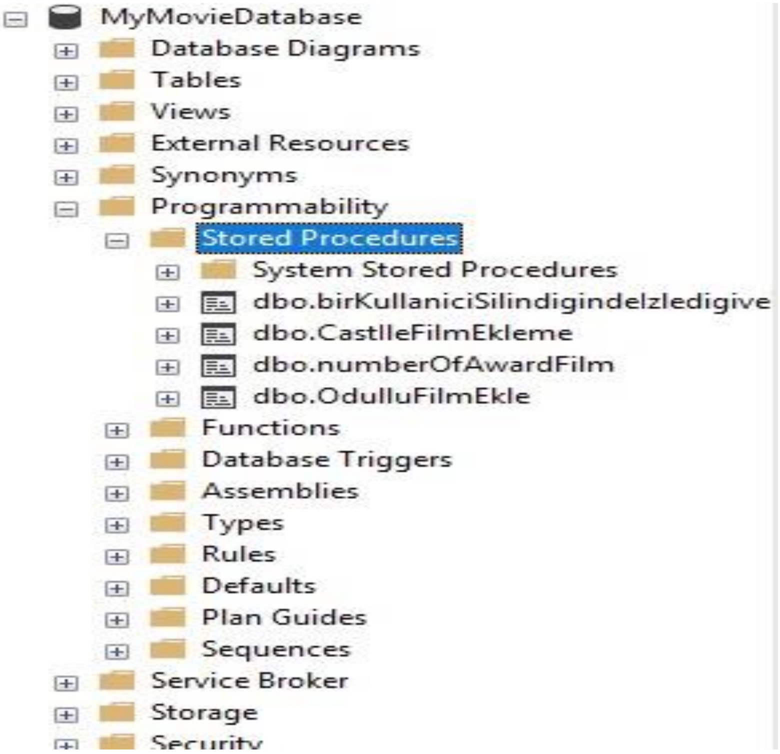
Procedure 4: Add cast with a movie.

*Can be found in STORED_PROCEDURE.sql

SCREENSHOTS

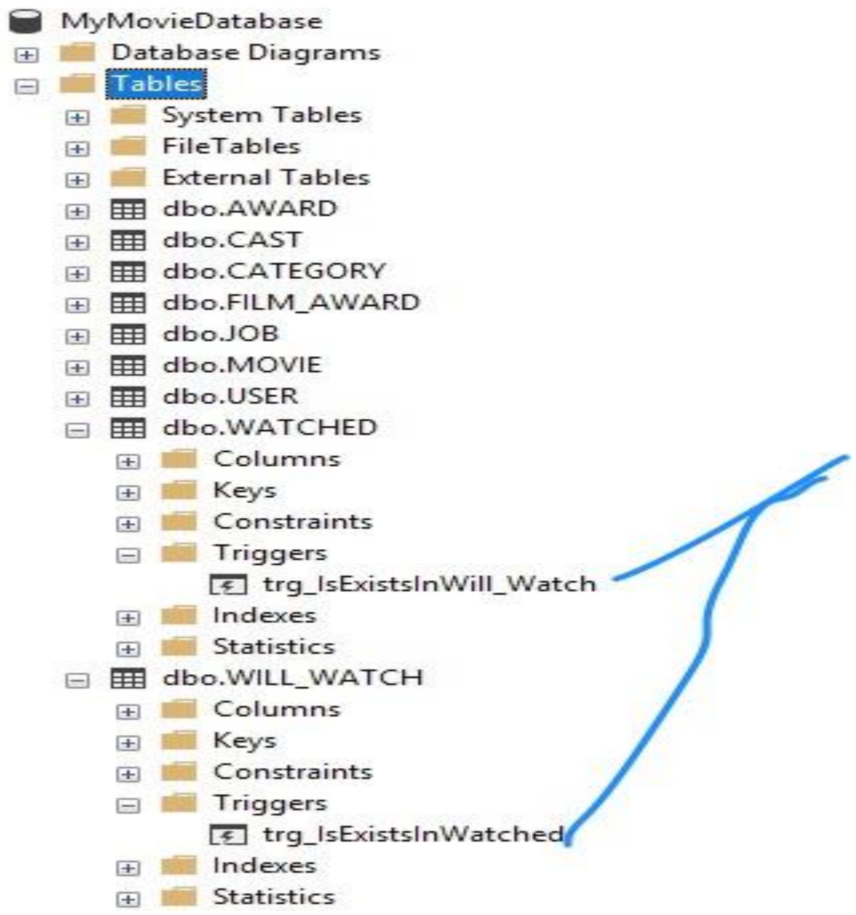


1 Diagram



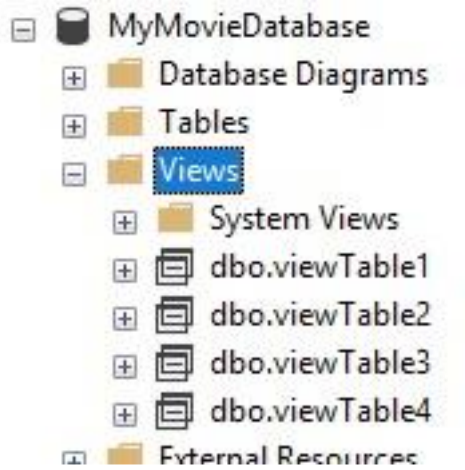
- MyMovieDatabase
  - Database Diagrams
  - Tables
  - Views
  - External Resources
  - Synonyms
  - Programmability
    - Stored Procedures
      - System Stored Procedures
      - dbo.birKullaniciSilindigindeIzledigive
      - dbo.CastIleFilmEkleme
      - dbo.numberOfAwardFilm
      - dbo.OdulluFilmEkle
    - Functions
    - Database Triggers
    - Assemblies
    - Types
    - Rules
    - Defaults
    - Plan Guides
    - Sequences
  - Service Broker
  - Storage
  - Security

2 Stored Procedures



```
MyMovieDatabase
  ⊞ Database Diagrams
  ⊟ Tables
      ⊞ System Tables
      ⊞ FileTables
      ⊞ External Tables
      ⊞ dbo.AWARD
      ⊞ dbo.CAST
      ⊞ dbo.CATEGORY
      ⊞ dbo.FILM_AWARD
      ⊞ dbo.JOB
      ⊞ dbo.MOVIE
      ⊞ dbo.USER
      ⊟ dbo.WATCHED
          ⊞ Columns
          ⊞ Keys
          ⊞ Constraints
          ⊟ Triggers
                trg_IsExistsInWill_Watch
          ⊞ Indexes
          ⊞ Statistics
      ⊟ dbo.WILL_WATCH
          ⊞ Columns
          ⊞ Keys
          ⊞ Constraints
          ⊟ Triggers
                trg_IsExistsInWatched
          ⊞ Indexes
          ⊞ Statistics
```

3 Triggers



```
⊟ MyMovieDatabase
    ⊞ Database Diagrams
    ⊞ Tables
    ⊟ Views
        ⊞ System Views
        ⊞ dbo.viewTable1
        ⊞ dbo.viewTable2
        ⊞ dbo.viewTable3
        ⊞ dbo.viewTable4
    ⊞ External Resources
```

4 Views

TOOLS

- Microsoft SQL Server Management System
- Github (Git Version Control System)