DROP TABLE MOVIES

DROP TABLE ACTORS

DROP TABLE SERIES

DROP TABLE COUNTRY

DROP TABLE DIRECTORS

DROP TABLE USER\_ACCOUNT

DROP TABLE PORTALS

DROP TABLE RATINGS

DROP TABLE PRODUCTION\_COMPANY

DROP TABLE USER\_MOVIELIST

DROP TABLE CATEGORY

DROP TABLE AGE\_LIMIT

DROP TABLE SUBTITLE\_LANGUAGE

CREATE TABLE ACTORS(

ActorID int IDENTITY(1,1),

SerieID int,

MovieID int,

CountryID int,

Vorname nvarchar(100) NOT NULL,

Nachname nvarchar(100) NOT NULL,

Geschlecht char(1) NOT NULL,

Bırthdate smalldatetime,

Email nvarchar(100),

City nvarchar(20),

CONSTRAINT pk\_ActorID PRIMARY KEY (ActorID),

CONSTRAINT chk\_Geschlect CHECK (Geschlecht='M' OR Geschlecht='F'),

)

CREATE TABLE MOVIES(

MovieID int IDENTITY(1,1),

MovieName nvarchar(100) NOT NULL,

MovieContent nvarchar(980) NOT NULL,

ReleaseDate smalldatetime,

MovieWebsite nvarchar(255),

CONSTRAINT pk\_MovieID PRIMARY KEY (MovieID),

CONSTRAINT fk\_MovieID FOREIGN KEY (MovieID)

REFERENCES MOVIES(MovieID)

)

CREATE TABLE SERIES(

SerieID int IDENTITY(1,1),

SerieName nvarchar(100) NOT NULL,

Season tinyint NOT NULL,

Partnummer smallint NOT NULL,

CONSTRAINT pk\_SerieID PRIMARY KEY (SerieID),

CONSTRAINT fk\_SerieID FOREIGN KEY (SerieID)

REFERENCES SERIES(SerieID)

)

CREATE TABLE COUNTRY(

CountryID int IDENTITY(1,1),

CountryName nvarchar(100) NOT NULL,

CountryCode int NOT NULL,

CONSTRAINT pk\_CountryID PRIMARY KEY (CountryID),

CONSTRAINT fk\_CountryID FOREIGN KEY (CountryID)

REFERENCES COUNTRY(CountryID)

)

CREATE INDEX CIndex

ON COUNTRY(CountryCode)

CREATE TABLE DIRECTORS (

DirectorID int IDENTITY(1,1),

MovieID int,

SerieID int,

CountryID int,

DirectorName nvarchar(100) NOT NULL,

DirectorVorname nvarchar(100) NOT NULL,

Awards nvarchar(100),

MovieNumber tinyint,

CONSTRAINT pk\_DirectorID PRIMARY KEY (DirectorID),

)

CREATE INDEX MIndex

ON DIRECTORS(MovieNumber)

CREATE TABLE USER\_ACCOUNT (

UserID int IDENTITY(1,1),

MovieID int,

PortalID int,

First\_Name nvarchar(100) NOT NULL,

Last\_Name nvarchar(100) NOT NULL,

EMail nvarchar(100) NOT NULL,

Password\_ nvarchar(100) NOT NULL,

PhoneNumber char(10) NOT NULL,

FilmList nvarchar(100),

CONSTRAINT pk\_UserID PRIMARY KEY (UserID),

CONSTRAINT fk\_UserID FOREIGN KEY (UserID)

REFERENCES USER\_ACCOUNT(UserID),

CONSTRAINT uc\_password UNIQUE (Password\_),

CONSTRAINT uc\_Name UNIQUE (First\_Name,Last\_Name)

)

CREATE TABLE PORTALS (

PortalID int IDENTITY(1,1),

PortalName nvarchar(200) NOT NULL,

PortalInfo nvarchar(500),

CONSTRAINT pk\_PortalID PRIMARY KEY (PortalID),

CONSTRAINT fk\_PortalID FOREIGN KEY (PortalID)

REFERENCES PORTALS(PortalID)

)

CREATE TABLE RATINGS(

RatingID int IDENTITY(1,1),

MovieID int,

UserID int,

RatingDate date DEFAULT GETDATE()

CONSTRAINT pk\_RatingID PRIMARY KEY (RatingID)

)

CREATE TABLE PRODUCTION\_COMPANY (

ProductionID int IDENTITY(1,1),

MovieID int,

SerieID int,

CompanyName nvarchar(100) NOT NULL

CONSTRAINT pk\_ProductionID PRIMARY KEY (ProductionID)

)

CREATE TABLE USER\_MOVIELIST (

UserID int IDENTITY(1,1),

MovieID int,

MovieName nvarchar(100),

UserName nvarchar(200),

CONSTRAINT pk\_UserID2 PRIMARY KEY (UserID,MovieID)

)

CREATE TABLE CATEGORY (

MovieGenreID int IDENTITY(1,1),

MovieID int,

SerieID int,

AgelimitID int,

GenreName varchar(225) NOT NULL,

CONSTRAINT pk\_MovieGenreID PRIMARY KEY (MovieGenreID),

CONSTRAINT fk\_MovieGenreID FOREIGN KEY (MovieGenreID)

REFERENCES CATEGORY(MovieGenreID)

)

CREATE TABLE AGE\_LIMIT (

AgelimitID int IDENTITY(1,1),

MovieID int,

MovieGenreID int,

Age tinyint NOT NULL ,

CONSTRAINT pk\_AgelimitID PRIMARY KEY (AgelimitID),

CONSTRAINT fk\_AgelimitID FOREIGN KEY (AgelimitID)

REFERENCES AGE\_LIMIT(AgelimitID),

CONSTRAINT chk\_Age CHECK (Age BETWEEN 0 AND 100)

)

CREATE TABLE SUBTITLE\_LANGUAGE (

LanguageID int IDENTITY(1,1),

MovieID int,

SerieID int,

LanguageName nvarchar(200) NOT NULL,

CONSTRAINT pk\_LanguageID PRIMARY KEY (LanguageID)

)

--SELECT \*

-- FROM MOVIES

--INNER JOIN ACTORS

--ON MOVIES.MovieID=ACTORS.MovieID;