using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
[Table("ADDRESSES")]  
public class Address {  
 [Column("ADDRESS\_ID")]  
 [Key, DatabaseGenerated(DatabaseGeneratedOption.**Identity**)]  
 public int Id { get; set; }  
  
 [Required, StringLength((8))]  
 [Column("POSTAL\_CODE")]  
 public string PostalCode { get; set; }  
   
 [Required,StringLength(100)]  
 [Column("LOCATION")]  
 public string Location { get; set; }  
   
 [Required, StringLength(100)]  
 [Column("STREET")]  
 public string Street { get; set; }  
   
 public State State { get; set; }  
   
 [Required]  
 [Column("STATE")]  
 public string StateCode { get; set; }  
   
 public Country Country { get; set; }  
   
 [Required]  
 [Column("COUNTRY")]  
 public string CountryName { get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
namespace Model.Entities;  
[Table("CINEMAS")]  
public class Cinema  
{  
 [Key,DatabaseGenerated(DatabaseGeneratedOption.**Identity**)]  
 [Column("CINEMA\_ID")]  
 public int Id { get; set; }  
   
 [Required,StringLength((100))]  
 [Column("LABEL")]  
 public string Label { get; set; }  
   
 public Address Address { get; set; }  
   
 [Required]  
 [Column("ADDRESS\_ID")]  
 public int AddressId { get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
[Table("E\_COUNTRIES")]  
public class Country {  
 [StringLength(100)]  
 [Key, Column("NAME")]  
 public string Name {get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
using Model.Entities;  
  
[Table(("HALLS"))]  
public class Hall{  
   
 [Key, DatabaseGenerated(DatabaseGeneratedOption.**Identity**)]  
 [Column("HALL\_ID")]  
 public int Id { get; set; }  
   
 [Required, Range(0, 500)]  
 [Column("CAPACITY")]  
 public int Capacity { get; set; }  
   
 [Required, StringLength(45)]  
 [Column("NAME")]  
 public string Name { get; set; }  
   
 [Required, StringLength(45)]  
 [Column("CODE")]  
 public string Code { get; set; }  
   
 *//relation* public Cinema Cinema { get; set; }  
   
 [Column("CINEMA\_ID")]  
 public int CinemaId { get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
[Table("MOVIES")]  
public class Movie {  
 [DatabaseGenerated(DatabaseGeneratedOption.**Identity**)]  
 [Key]  
 [Column("MOVIE\_ID")]  
 public int Id { get; set; }  
   
 [Required]  
 [Column("TITLE")]  
 [StringLength(100)]  
 public string Title { get; set; }  
   
 [Required]  
 [Column("DIRECTOR")]  
 [StringLength(50)]  
 public string Director { get; set; }  
   
 [Required]  
 [Column("LENGTH")]  
 [MinLength(0)]  
 [MaxLength(900)]  
 *// [Range(0, 900)]* public int Length { get; set; }  
   
 [Required]  
 [Column("SHORT\_DESCRIPTION")]  
 [StringLength(400)]  
 public string ShortDescription { get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
[Table("MOVIE\_CREWS")]  
public class Person {  
   
 [Key, DatabaseGenerated(DatabaseGeneratedOption.**Identity**)]  
 [Column("PERSON\_ID")]  
 public int Id { get; set; }  
   
 [Required, StringLength(45)]  
 [Column("FIRST\_NAME")]  
 public string FirstName { get; set; }  
   
 [Required, StringLength(45)]  
 [Column("LAST\_NAME")]  
 public string LastName { get; set; }  
}

using System.ComponentModel.DataAnnotations;  
using System.ComponentModel.DataAnnotations.Schema;  
  
[Table("E\_STATES")]  
public class State {  
 [StringLength(100)]  
 [Key, Column("NAME")]  
 public string Name {get; set; }  
}

using Microsoft.EntityFrameworkCore;  
using Model.Entities;  
  
namespace Model.Configurations;  
  
public class MovieContext : DbContext {  
 public MovieContext(DbContextOptions<MovieContext> options) : base(options) { }  
   
 public DbSet<Movie> Movies{ get; set; }  
 public DbSet<Person> People { get; set; }  
 public DbSet<State> States { get; set; }  
 public DbSet<Country> Countries { get; set; }  
 public DbSet<Address> Addresses { get; set; }  
 public DbSet<Cinema> Cinemas { get; set; }  
 public DbSet<Hall> Halls { get; set; }  
  
 protected override void OnModelCreating(ModelBuilder builder)  
 {  
 builder.Entity<Cinema>()  
 .HasOne(c => c.Address)  
 .WithOne()  
 .HasForeignKey<Cinema>(c => c.AddressId);  
   
 *//Address* builder.Entity<Address>()  
 .HasOne(a => a.Country)  
 .WithMany()  
 .HasForeignKey(a => a.CountryName);  
  
 builder.Entity<Address>()  
 .HasOne(a => a.State)  
 .WithMany()  
 .HasForeignKey(a => a.StateCode);  
   
 *//hall* builder.Entity<Hall>()  
 .HasOne(h => h.Cinema)  
 .WithMany()  
 .HasForeignKey(h => h.CinemaId);  
 }  
  
}

INSERT INTO E\_COUNTRIES(NAME)  
VALUES ('Austria');  
INSERT INTO E\_COUNTRIES(NAME)  
VALUES ('France');  
INSERT INTO E\_COUNTRIES(NAME)  
VALUES ('Germany');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Burgenland');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Kärnten');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Niederösterreich');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Oberösterreich');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Salzburg');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Steiermark');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Tirol');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Vorarlberg');  
INSERT INTO E\_STATES(NAME)  
VALUES ('Wien');  
  
  
INSERT INTO ADDRESSES (ADDRESS\_ID, POSTAL\_CODE, LOCATION, STREET,STATE, COUNTRY)  
VALUES (1, '3500', 'Krems a.d. Donau','Gewerbestrasse 21', 'Niederösterreich', 'Austria');  
INSERT INTO ADDRESSES (ADDRESS\_ID, POSTAL\_CODE, LOCATION, STREET,STATE, COUNTRY)  
VALUES (3, '3500', 'Krems a.d. Donau', 'Dr Karl Dorrek Strasse 30', 'Niederösterreich', 'Austria');  
INSERT INTO CINEMAS (CINEMA\_ID, LABEL, ADDRESS\_ID)  
VALUES (1,'Cinemaplexx Krems', 1);  
INSERT INTO CINEMAS (CINEMA\_ID, LABEL, ADDRESS\_ID)  
VALUES (3, 'Kino im Kesselhaus', 3);  
INSERT INTO HALLS (HALL\_ID, CAPACITY, NAME, CODE, CINEMA\_ID)  
VALUES (1, 50, 'Saal A', 'krems\_A', 1);  
INSERT INTO HALLS (HALL\_ID, CAPACITY, NAME, CODE, CINEMA\_ID)   
VALUES (2, 120, 'Saal B', 'krems\_b', 1);  
INSERT INTO HALLS (HALL\_ID, CAPACITY, NAME, CODE, CINEMA\_ID)  
VALUES (3, 40, 'Kesselsaal', 'KL-34', 3);

Ein Bild, das Text, Software, Multimedia-Software, Screenshot enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Software, Multimedia-Software, Grafiksoftware enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Software, Multimedia-Software, Screenshot enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung