## Student System

Your task is to create a database for the **StudentSystem**, using the **EF Core Code First** approach. It should look like this:



### Constraints

Your **namespaces** should be:

* P01\_StudentSystem – for your **Startup** class, if you have one
* P01\_StudentSystem.Data – for your **DbContext**
* P01\_StudentSystem.Data.Models – for your models

Your **models** should be:

* StudentSystemContext – your **DbContext**
* Student
  + **StudentId**
  + **Name** – up to 100 characters, unicode
  + **PhoneNumber** – exactly 10 characters, not unicode, not required
  + **RegisteredOn**
  + **Birthday** – not required
* Course
  + **CourseId**
  + **Name** – up to 80 characters, unicode
  + **Description** – unicode, not required
  + **StartDate**
  + **EndDate**
  + **Price**
* Resource
  + **ResourceId**
  + **Name** – up to 50 characters, unicode
  + **Url** – not unicode
  + **ResourceType** – **enum**, can be **Video**, **Presentation**, **Document** or **Other**
  + **CourseId**
* Homework
  + **HomeworkId**
  + **Content** – **string**, linking to a file, not unicode
  + **ContentType** - **enum**, can be **Application**, **Pdf** or **Zip**
  + **SubmissionTime**
  + **StudentId**
  + **CourseId**
* StudentCourse – **mapping** between **Students** and **Courses**

Table relations:

* **One student** can have **many Courses**
* **One student** canhave **many Homeworks**
* **One course** can have **many Students**
* **One course** can have **many Resources**
* **One course** can have **many Homeworks**

You will need a constructor, accepting **DbContextOptions** to test your solution in **Judge**!

public class Course

{

[Key]

public int CourseId { get; set; }

[MaxLength(80)]

public string Name { get; set; } = null!;

public string? Description { get; set; }

[Required]

public DateTime StartDate { get; set; }

public DateTime EndDate { get; set; }

public decimal Price { get; set; }

public virtual ICollection<Resource> Resources { get; set; }

public virtual ICollection<Homework> Homeworks { get; set; }

public virtual ICollection<StudentCourse> StudentsCourses { get; set; }

}

public class Homework

{

[Key]

public int HomeworkId { get; set; }

[Column(TypeName = "VARCHAR")]

public string? Content { get; set; }

public ContentType ContentType { get; set; }

[Required]

public DateTime SubmissionTime { get; set; }

public int StudentId { get; set; }

[ForeignKey(nameof(StudentId))]

public virtual Student Student { get; set; }

public int CourseId { get; set; }

[ForeignKey(nameof(CourseId))]

public Course Course { get; set; }

}

public class Resource

{

[Key]

public int ResourceId { get; set; }

[MaxLength(50)]

public string Name { get; set; } = null!;

[Required]

[Column(TypeName = "VARCHAR")]

public string Url { get; set; } = null!;

public ResourceType ResourceType { get; set; }

public int CourseId { get; set; }

[ForeignKey(nameof(CourseId))]

public virtual Course Course { get; set; }

}

public class StudentCourse

{

public int StudentId { get; set; }

[ForeignKey(nameof(StudentId))]

public virtual Student Student { get; set; }

public int CourseId { get; set; }

[ForeignKey(nameof(CourseId))]

public virtual Course Course { get; set; }

}

public class Student

{

[Key]

public int StudentId { get; set; }

[MaxLength(100)]

public string Name { get; set; } = null!;

[MinLength(10)]

[MaxLength(10)]

[Column(TypeName = "VARCHAR")]

public string? PhoneNumber { get; set; }

[Required]

public DateTime RegisteredOn { get; set; }

public DateTime? Birthday { get; set; }

public virtual ICollection<Homework> Homeworks { get; set; }

public virtual ICollection<StudentCourse> StudentsCourses { get; set; }

}

public enum ContentType

{

Application,

Pdf,

Zip

}

public enum ResourceType

{

Video,

Presentation,

Document,

Other

}

public class StudentSystemContext : DbContext

{

//public StudentSystemContext(DbContextOptions dbContextOptions) : base(dbContextOptions)

//{}

public DbSet<Student> Students { get; set; }

public DbSet<Course> Courses { get; set; }

public DbSet<StudentCourse> StudentsCourses { get; set; }

public DbSet<Resource> Resources { get; set; }

public DbSet<Homework> Homeworks { get; set; }

string connectionstring = "Server=192.168.88.40 ,1434; Database = StudentSystem; User Id = sa; Password = password;";

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer(connectionstring);

}

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<StudentCourse>()

.HasKey(sc => new { sc.StudentId, sc.CourseId });

}

}