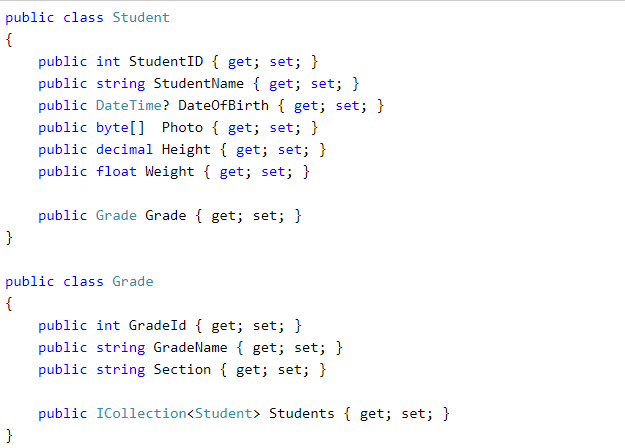
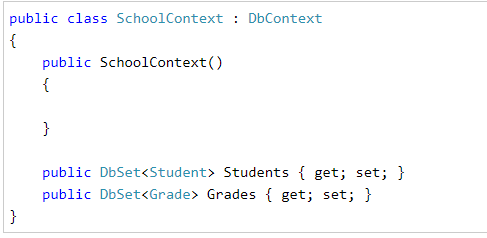
Entity Framework

What is an Entity in Entity Framework?

An entity in Entity Framework is a class that maps to a database table. EF API maps each entity to a table and each property to a column in the database.

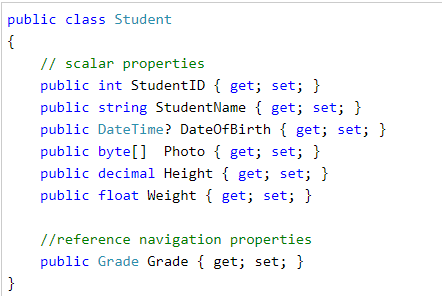


There classes be inculed as DbSet<TEntity> type property in the DbContext



Scalar Property (thuộc tính vô hướng)

The primitive type properties are called scalar properties. Each scalar property map to a column in the database table. For example, StudentID, StudentName, DateOfBirth, Photo, Height, Weight are the scalar properties in the Student entity class.



Navigation Property (thuộc tính điều hướng)

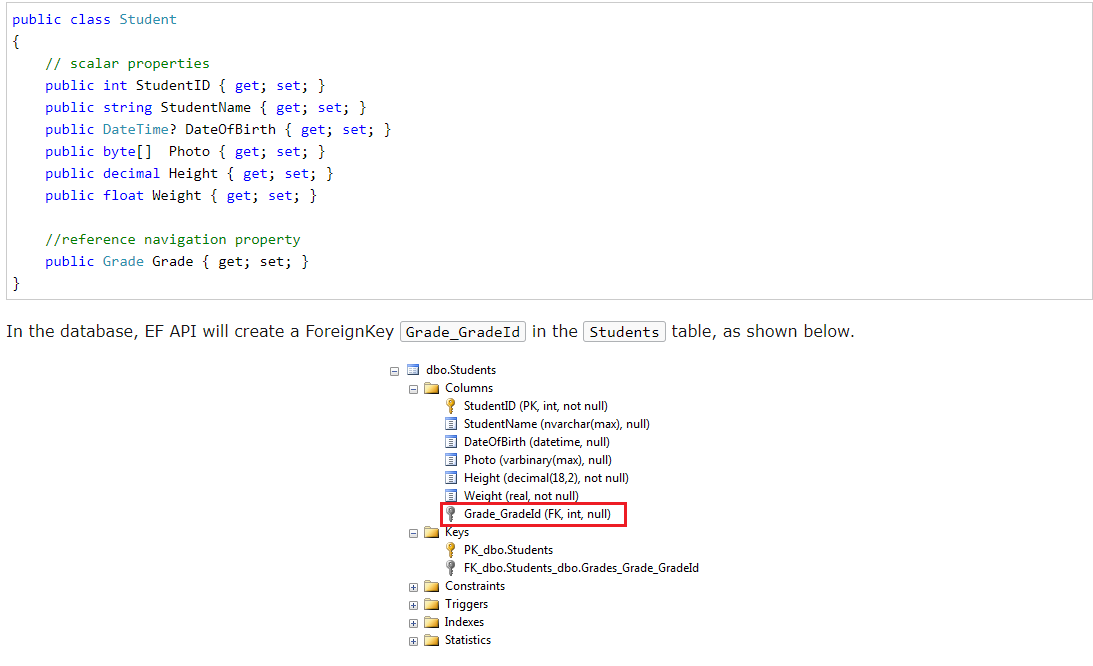
The navigation property presents a relationship to other entity. (có nghĩa là nó sẽ điều hướng tới một entity khác nên gọi là thuộc tính điều hướng)

There are two types of navigation properties: Reference Navigation and Collection Navigation

* Reference Navigation Property

If an entity includes a property of other entity type, it is called a Reference Navigation Property.

EF API does will create a ForeignKey column in table for the navigation properties that points to a PrimaryKey of another table in the database. For example, Grade are reference navigaion properties in the following Student entity class



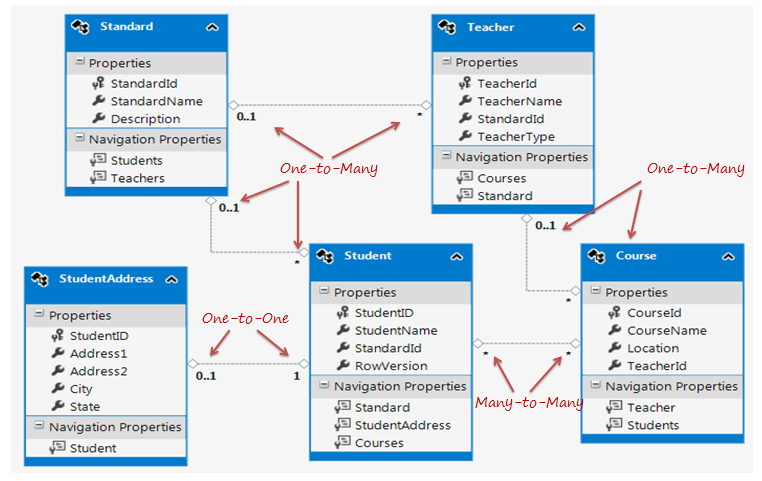
Nếu ta không định nghĩa cho trước scalar property đóng vai trò làm ForeignKey cho thằng Navigation Property là Grade thì trong database nó sẽ tự tạo một cột và đặt mặc định với tên là Grade\_GradeId để giữ id của các thằng trong bảng Grade.

* Collection Navigation Property

If an entity includes a property of generic collection of an entity type, it is called a collection navigation property.

EF API does not create any column for the collection navigation property in the related tbale of an entity, but it creates a column in the table of an entity of generic collection.

Entity Relationships

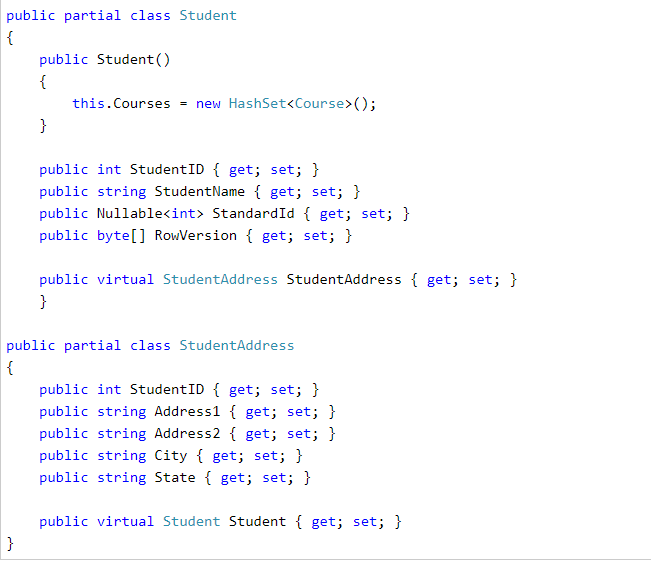


Entity Framework supports three types of relationships

1-1, 1-n,n-n

* One-toOne Relationship:

As you can see in the above figure, Student and StudentAddress have a One-to-One relationships. Lúc này cả 2 thằng sẽ sử dụng chung Id để có thể tự tìm đến nhau.



Như hình trên ta thấy cả 2 entity là Student và StudentAddress cùng sử dụng chung StudentID và navigation property có thể dựa trên FK là StudentID để lấy được dữ liệu.

* One-To-Many RelationShip

The Standard and Teacher entities have a One-to-Many relationship. This means that Standard can have many Teachers whereas Teacher can associate with only one Standard (Một tiêu chuẩn có thể có nhiều giáo viên đạt đến và giữ nó, mỗi giáo viên thì sẽ chỉ có duy nhất một tiêu chuẩn).

Để thể hiện được mối quan hệ này, thằng Standard sẽ có một collection navigation property Teachers, cái này nói lên rắng là một Standard có thể có một danh sách Teacher có mối quan hệ với một Standard, VÌ vậy thằn Teacher chứa StandardId làm FK

