

```

public class Project
{
    // 1- Id PK with identity of (10,10)
    // (Primary Key) هو المفتاح الأساسي 'Id' هذا يعني أن
    // وقيمته تزيد تلقائيًا، تبدأ من 10 وتزيد بمقدار 10 لكل سجل جديد.
    public int Id { get; set; }

    // 2- Name prop Map it to varchar(50), has default value of "OurProject", Required column
    // يمثل اسم المشروع (string) هو حقل نصي 'Name'.
    // ويبلغ أقصى طول لها 50 حرفًا (Required) يجب أن تكون قيمته إلزامية.
    // القيمة الافتراضية له هي "OurProject".
    public string Name { get; set; }

    // 3- Cost column make it Money datatype, make the Range between 500000 and 3500000.
    // Money هو حقل يمثل التكلفة، ونوعه هو 'Cost'.
    // يجب أن تكون قيمته ضمن النطاق المحدد (بين 500,000 و 3,500,000)
    public decimal Cost { get; set; }
}

```

---

```

using Microsoft.EntityFrameworkCore;

```

```

using Microsoft.EntityFrameworkCore.Metadata.Builders;

```

```

public class ProjectConfiguration : IEntityTypeConfiguration<Project>
{
    public void Configure(EntityTypeBuilder<Project> builder)

```

```
{  
    // إعدادات 'Id'  
  
    builder.HasKey(p => p.Id); // كمفتاح أساسي Id تعيين (Primary Key)  
  
    builder.Property(p => p.Id)  
        .HasIdentity<int>(seed: 10, increment: 10); // (identity) إعداد الزيادة التلقائية  
  
    // إعدادات 'Name'  
  
    builder.Property(p => p.Name)  
        .HasMaxLength(50) // varchar(50) تعيين أقصى طول لـ  
        .IsRequired() // (NOT NULL) جعله إلزاميًا  
        .HasDefaultValue("OurProject"); // تعيين القيمة الافتراضية  
  
    // إعدادات 'Cost'  
  
    // SQL Server في `money` بنوع `decimal` ، يمكن ربط EF Core في  
    builder.Property(p => p.Cost)  
        .HasColumnType("money");  
  
    // ، يتم التحقق منه عادةً على مستوى (Range) بالنسبة للنطاق  
    // "constraints" أو باستخدام (application level) الكود البرمجي  
    // في قاعدة البيانات  
}  
}
```

---