

Entity framework Core

<https://learn.microsoft.com/en-us/ef/core/managing-schemas/scaffolding/?tabs=dotnet-core-cli>

Chuẩn bị công cụ

\$ dotnet tool install --global dotnet-ef

NuGet package: (remember to build after install)


- Microsoft.EntityFrameworkCore.Design
- Microsoft.EntityFrameworkCore.SqlServer



Reverse Engineering (Right click > Open in Terminal)

```
$ dotnet ef dbcontext scaffold "Data Source=.\SqlExpress;  
Trusted_Connection=Yes; Initial Catalog=SuperShop;  
TrustServerCertificate=True" Microsoft.EntityFrameworkCore.SqlServer
```

```
$ dotnet ef dbcontext scaffold "Data Source=localhost; Initial  
Catalog=demoshop; TrustServerCertificate=True; User Id=sa;  
Password=SqlServer@123" Microsoft.EntityFrameworkCore.SqlServer
```



Một số lưu ý

- Hàm **OnConfiguring** chứa thông tin nhạy cảm, không nên tồn tại
 - Sửa bằng cách thêm đối số **--no-onconfiguring** nhằm loại bỏ trong file Context tạo ra
 - Sử dụng partial class để tạo ra hàm OnConfiguring này riêng
 - Trong đó sử dụng localSettings để nạp connection string
- Khi có thay đổi về CSDL cần cập nhật lại, sử dụng đối số **--force**
 - Các lớp sinh ra có thể được đặt trong thư mục **Models** với đối số **--output-dir Models**
- Để giới hạn chỉ tạo ra entity từ một số bảng nhất định, chỉ định rõ với đối số **--table**
 - Ví dụ: `dotnet ef dbcontext scaffold ... --table Artist --table Album --table Employee --table Product`
- Để thay đổi thư mục chứa context, dùng đối số **--context-dir Data**

Khởi tạo kết nối tới CSDL

```
var context = new TestDbContext();  
if (context.Database.CanConnect()){  
  
}
```



Select

```
var categories = context.Categories.ToList();
```



Insert

```
var category = new Category() { Name = "Test" };
```

```
context.Categories.Add(category);
```

```
context.SaveChanges();
```

```
var id = category.Id;
```



Delete

```
var categories = context.Categories.ToList();
```

```
var selectedCategory = categories[0];
```

```
context.Categories.Remove(selectedCategory);
```

```
context.SaveChanges();
```



Update

```
var categories = context.Categories.ToList();
```

```
var selectedCategory = categories[0];
```

```
selectedCategory.Name = "SuperArt";
```

```
context.SaveChanges();
```

UI: Add **INotifyPropertyChanged**



Phân trang: lấy tổng số sản phẩm

Để tránh 2 roundtrip ta có thể truy vấn tối ưu như sau

```
var results = query.OrderBy(p => p.Name)
                    .Select(p => new
                    {
                        Person = new PersonResult { Name = p.Name },
                        TotalCount = query.Count()
                    })
                    .Skip(skipRows).Take(pageSize)
                    .ToArray(); // query is executed once, here

var totalCount = results.FirstOrDefault()?.TotalCount ?? query.Count();
var people = results.Select(r => r.Person).ToArray();
```