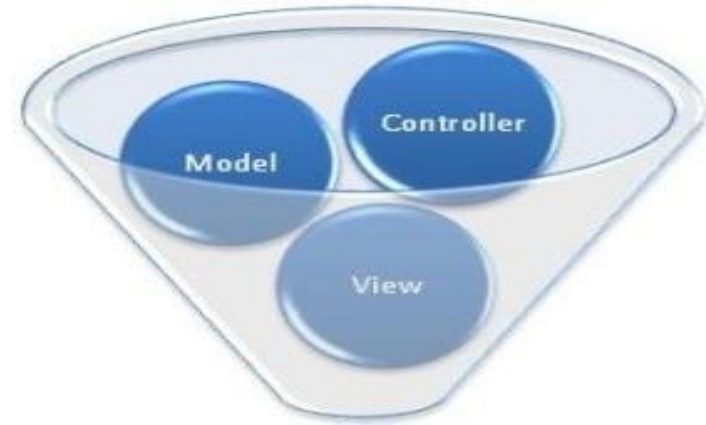


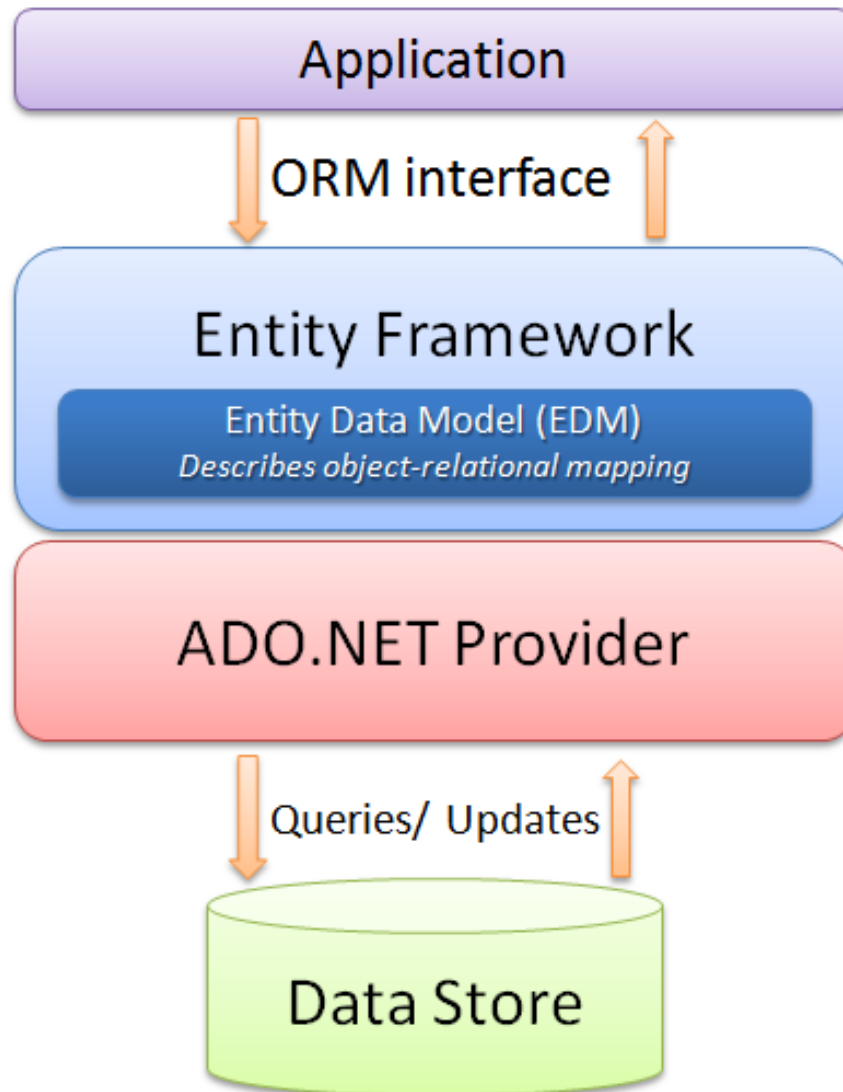
Microsoft  
**ASP.net MVC 5**

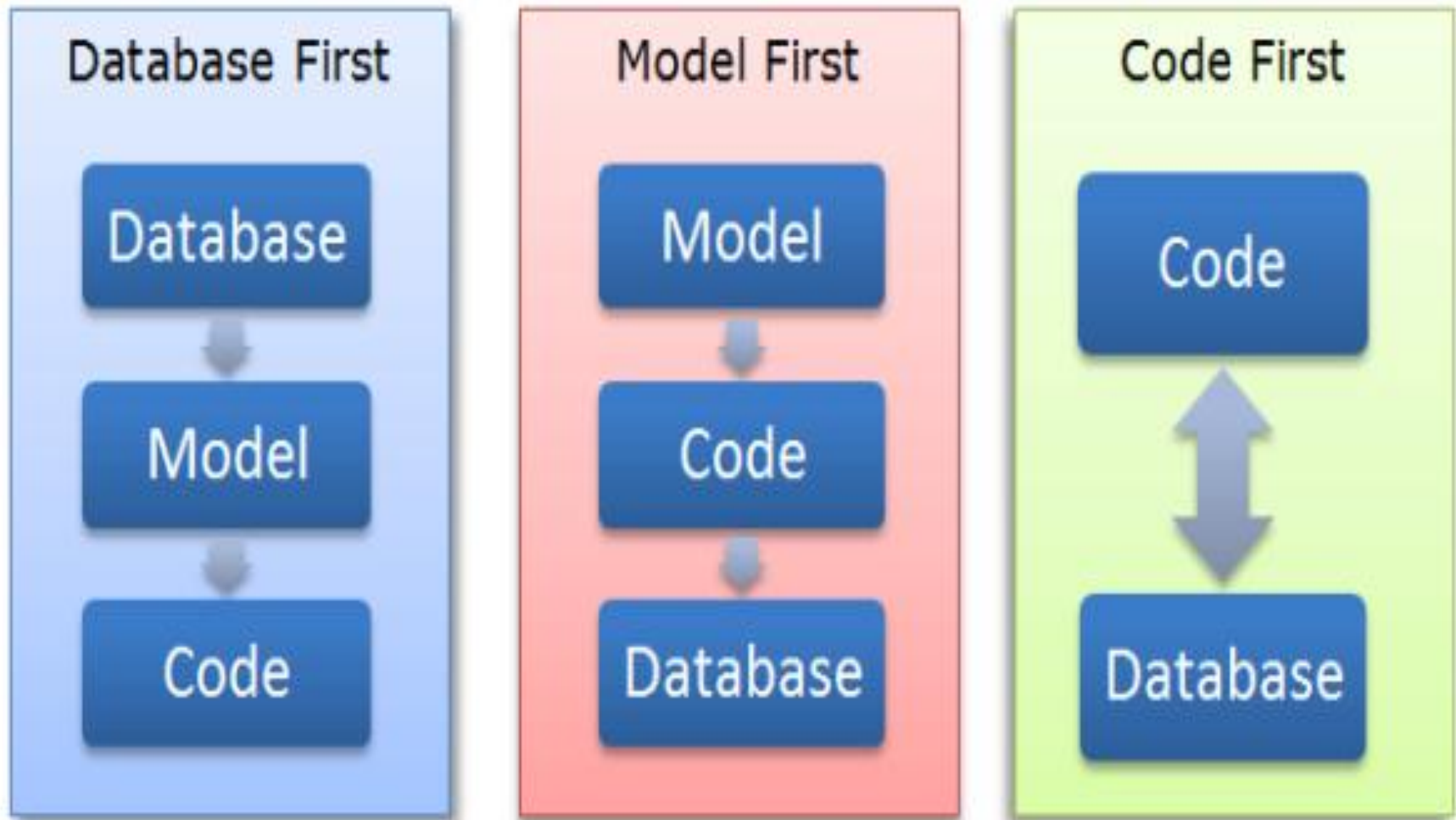


**ENTITY FRAMEWORK**

ThS. Nguyễn Nghiệm  
0913.745.789  
nghiemn@fpt.edu.vn  
songlong2k@gmail.com

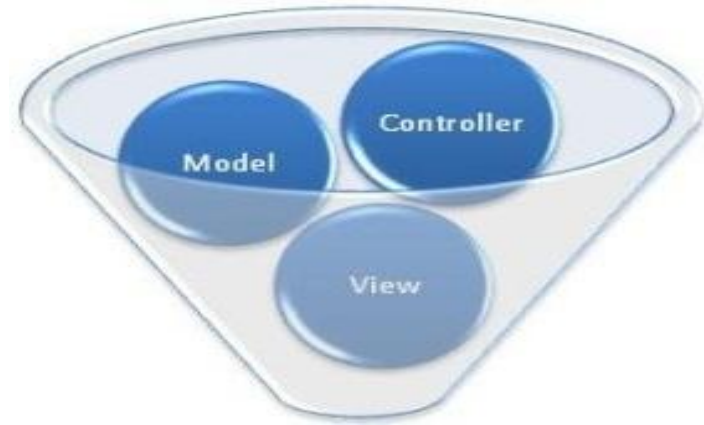
- ❑ Kiến trúc tổ chức
- ❑ Các mô hình lập trình
- ❑ Mô hình Database First
- ❑ EF API
- ❑ Mô hình Code First
  - ✍ Xây dựng Entity
  - ✍ Xây dựng Database Context
  - ✍ Khởi tạo dữ liệu
  - ✍ Thay đổi thuộc tính





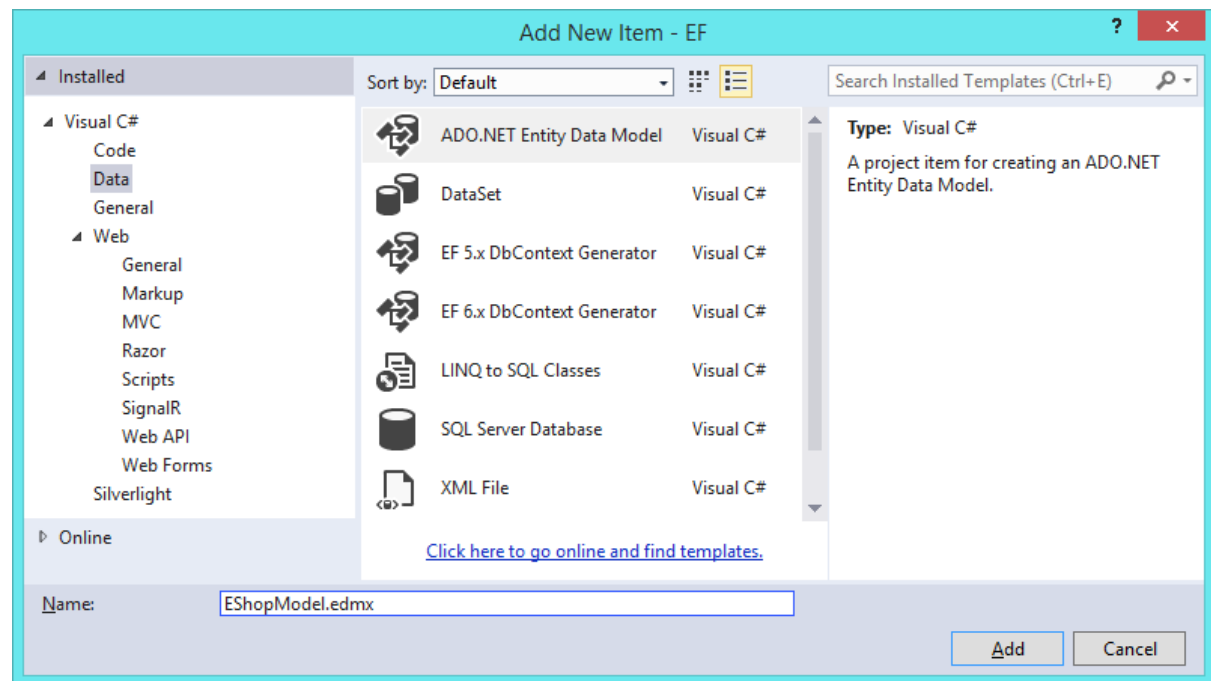
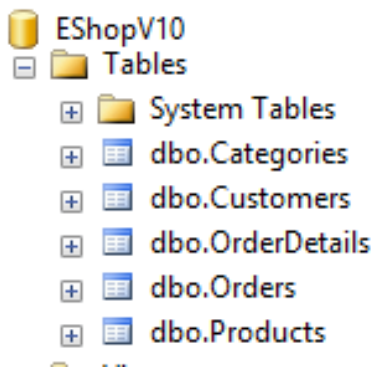
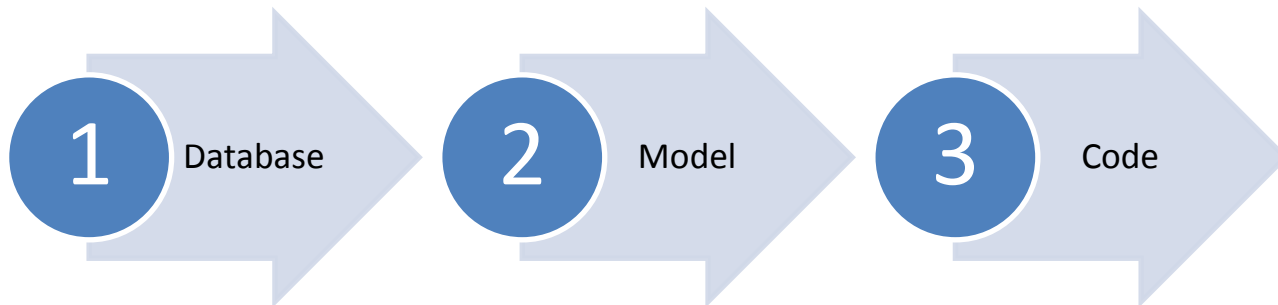


Microsoft  
**ASP.net MVC 5**



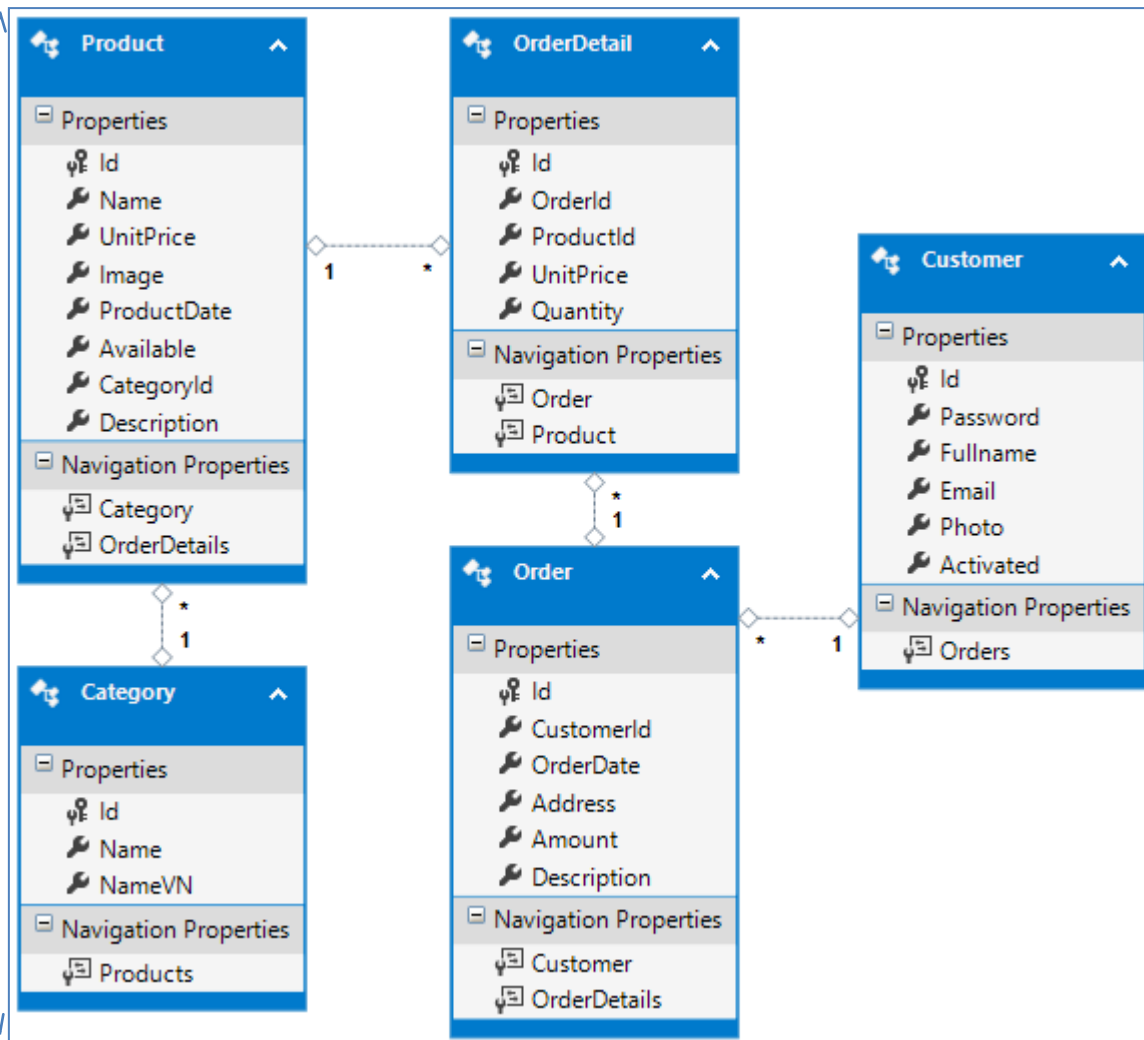
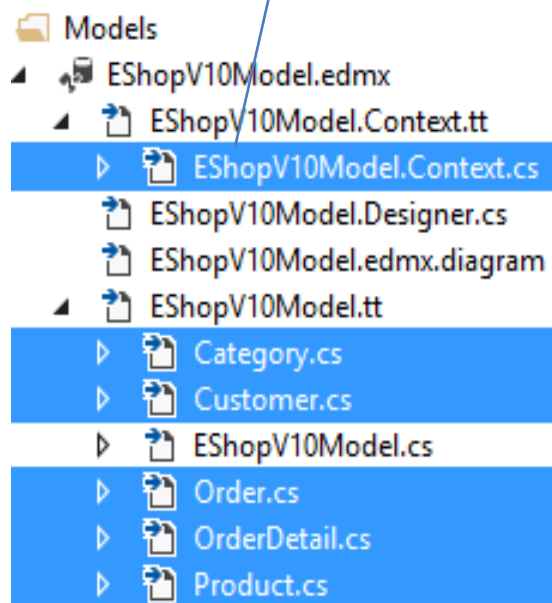
**DATABASE FIRST**

ThS. Nguyễn Nghiệm  
0913.745.789  
nghiemn@fpt.edu.vn  
songlong2k@gmail.com



## Entity

## Database Context



```
public partial class EShopV10 : DbContext
{
    public EShopV10()
        : base("name=EShopV10")
    {
    }

    public DbSet<Category> Categories { get; set; }
    public DbSet<Customer> Customers { get; set; }
    public DbSet<OrderDetail> OrderDetails { get; set; }
    public DbSet<Order> Orders { get; set; }
    public DbSet<Product> Products { get; set; }
}
```

```
<connectionStrings>
    <add name="EShopV10" connectionString="..."
        providerName="System.Data.EntityClient" />
</connectionStrings>
```



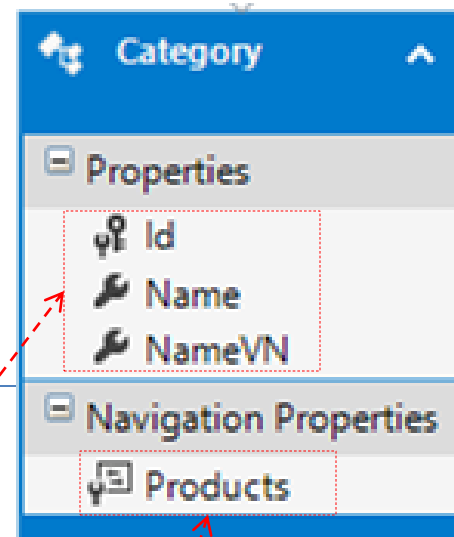
- ❑ Tên thực thể  $\Leftrightarrow$  Tên bảng
- ❑ Tên thuộc tính  $\Leftrightarrow$  Tên cột
- ❑ Thực thể kết hợp 1-Nhiều

```
public partial class Category
```

```
{  
    public Category()  
    {  
        this.Products = new HashSet<Product>();  
    }  
}
```

```
public int Id { get; set; }  
public string Name { get; set; }  
public string NameVN { get; set; }
```

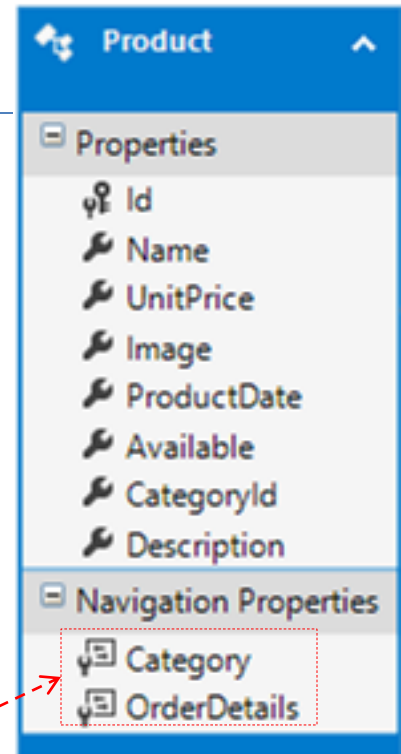
```
public virtual ICollection<Product> Products { get; set; }
```



```
public partial class Product
{
    public Product()
    {
        this.OrderDetails = new HashSet<OrderDetail>();
    }

    public int Id { get; set; }
    public string Name { get; set; }
    public double UnitPrice { get; set; }
    public string Image { get; set; }
    public System.DateTime ProductDate { get; set; }
    public bool Available { get; set; }
    public int CategoryId { get; set; }
    public string Description { get; set; }

```



Product

Properties

- Id
- Name
- UnitPrice
- Image
- ProductDate
- Available
- CategoryId
- Description

Navigation Properties

- Category
- OrderDetails

```
public virtual Category Category { get; set; }
public virtual ICollection<OrderDetail> OrderDetails { get; set; }
}
```

## ❑ Tạo đối tượng db context

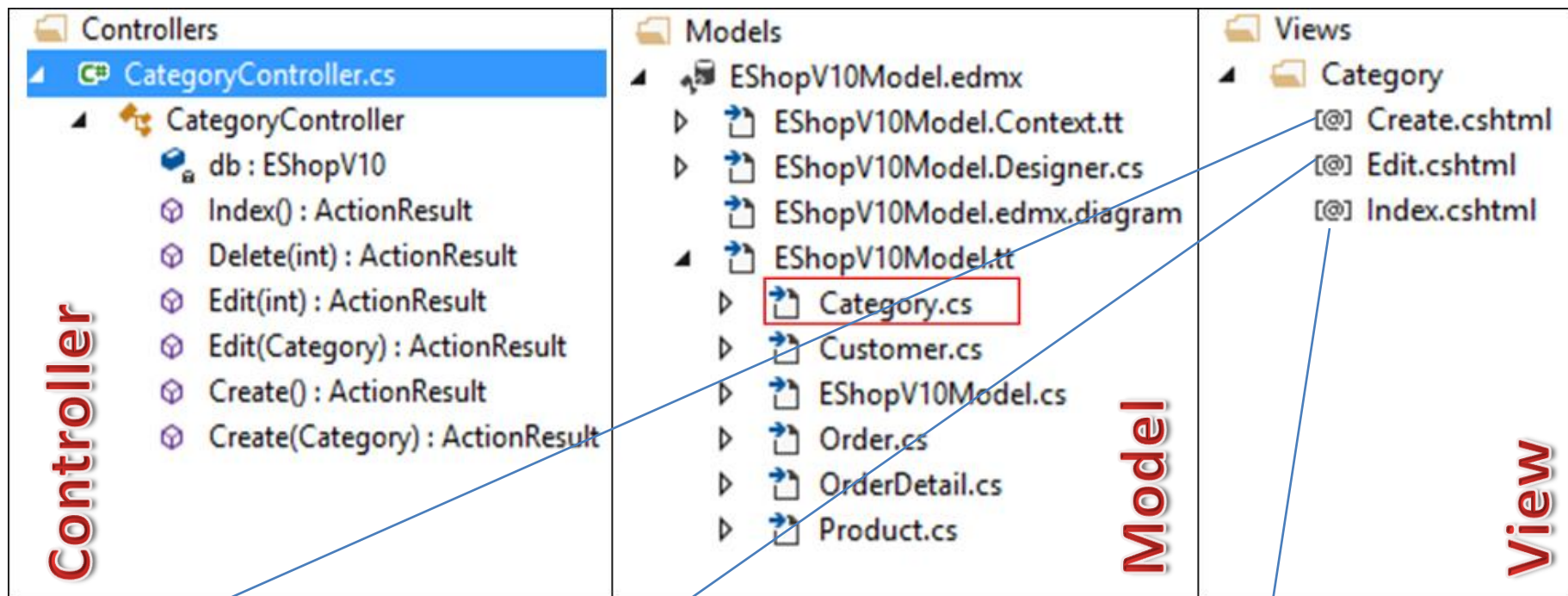
```
var db = new EShopV10();
```

## ❑ Thao tác & truy vấn thực thể

|                                 |  |
|---------------------------------|--|
| Thêm mới thực thể               | db.Categories. <b>Add</b> (category)                             |
| Cập nhật thông tin của thực thể | db. <b>Entry</b> (category). <b>State</b> = EntityState.Modified |
| Xóa thực thể                    | db.Categories. <b>Remove</b> (category)                          |
| Truy vấn một thực thể theo mã   | var entity = db.Categories. <b>Find</b> (id)                     |
| Truy vấn tất cả các thực thể    | var list = db. <b>Categories</b>                                 |

## ❑ Lưu sự thay đổi

```
db.SaveChanges()
```



**Name**

**NameVN**

Create

**Name**

**NameVN**

Save

| Id   | Name    | NameVN            |               |
|------|---------|-------------------|---------------|
| 1000 | Watches | Đồng hồ đeo tay   | Edit : Delete |
| 1001 | Laptops | Máy tính xách tay | Edit : Delete |
| 1002 | Cameras | Máy ảnh           | Edit : Delete |

```
// GET:/Category/Create
public ActionResult Create()
{
    return View();
}

// POST:/Category/Create
[HttpPost]
public ActionResult Create(Category model)
{
    // Thêm mới
    db.Categories.Add(model);
    db.SaveChanges();
    return RedirectToAction("Index");
}
```

**Name**

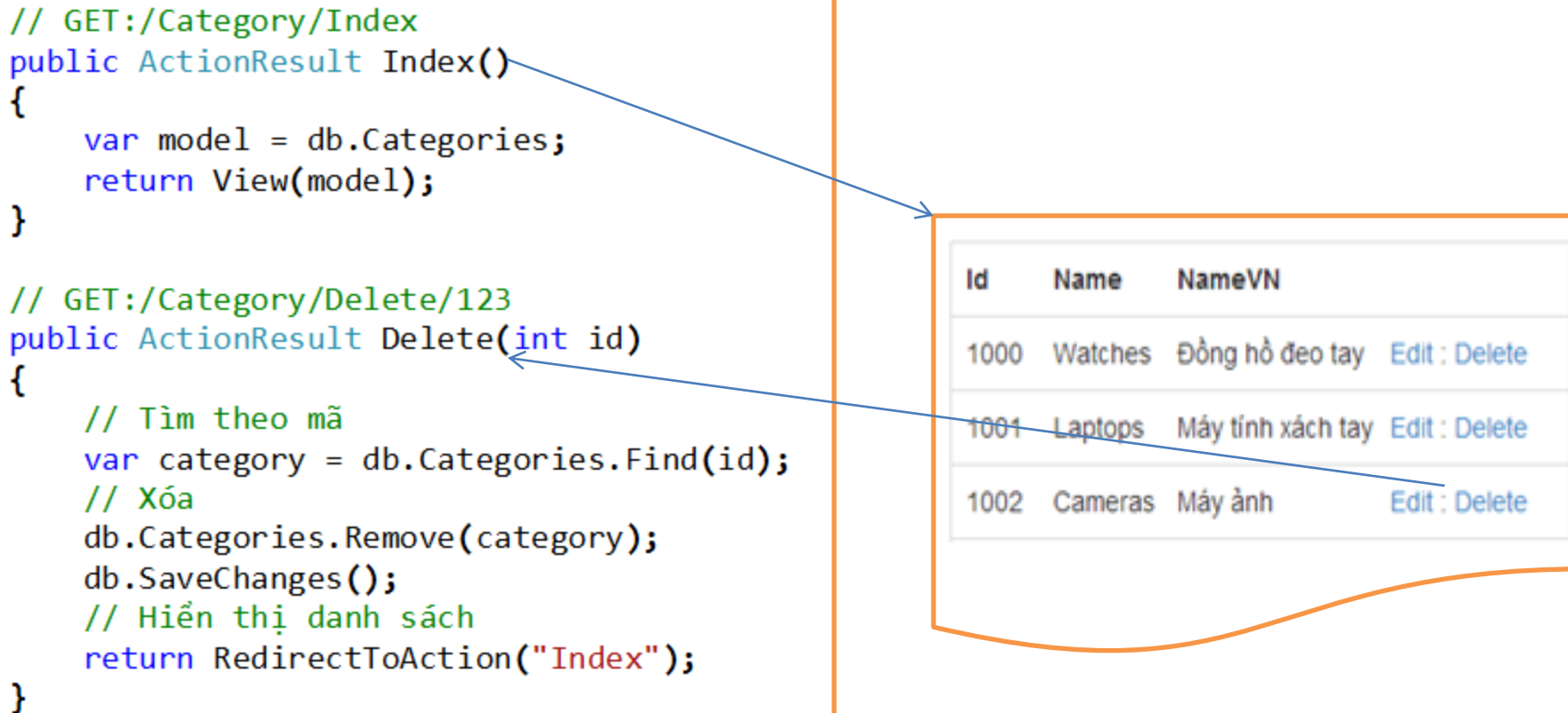
**NameVN**

Create



```
// GET:/Category/Index
public ActionResult Index()
{
    var model = db.Categories;
    return View(model);
}

// GET:/Category/Delete/123
public ActionResult Delete(int id)
{
    // Tìm theo mã
    var category = db.Categories.Find(id);
    // Xóa
    db.Categories.Remove(category);
    db.SaveChanges();
    // Hiển thị danh sách
    return RedirectToAction("Index");
}
```



| Id   | Name    | NameVN            |   |
|------|---------|-------------------|---|
| 1000 | Watches | Đồng hồ đeo tay   | <a href="#">Edit</a> : <a href="#">Delete</a> |
| 1001 | Laptops | Máy tính xách tay | <a href="#">Edit</a> : <a href="#">Delete</a> |
| 1002 | Cameras | Máy ảnh           | <a href="#">Edit</a> : <a href="#">Delete</a> |

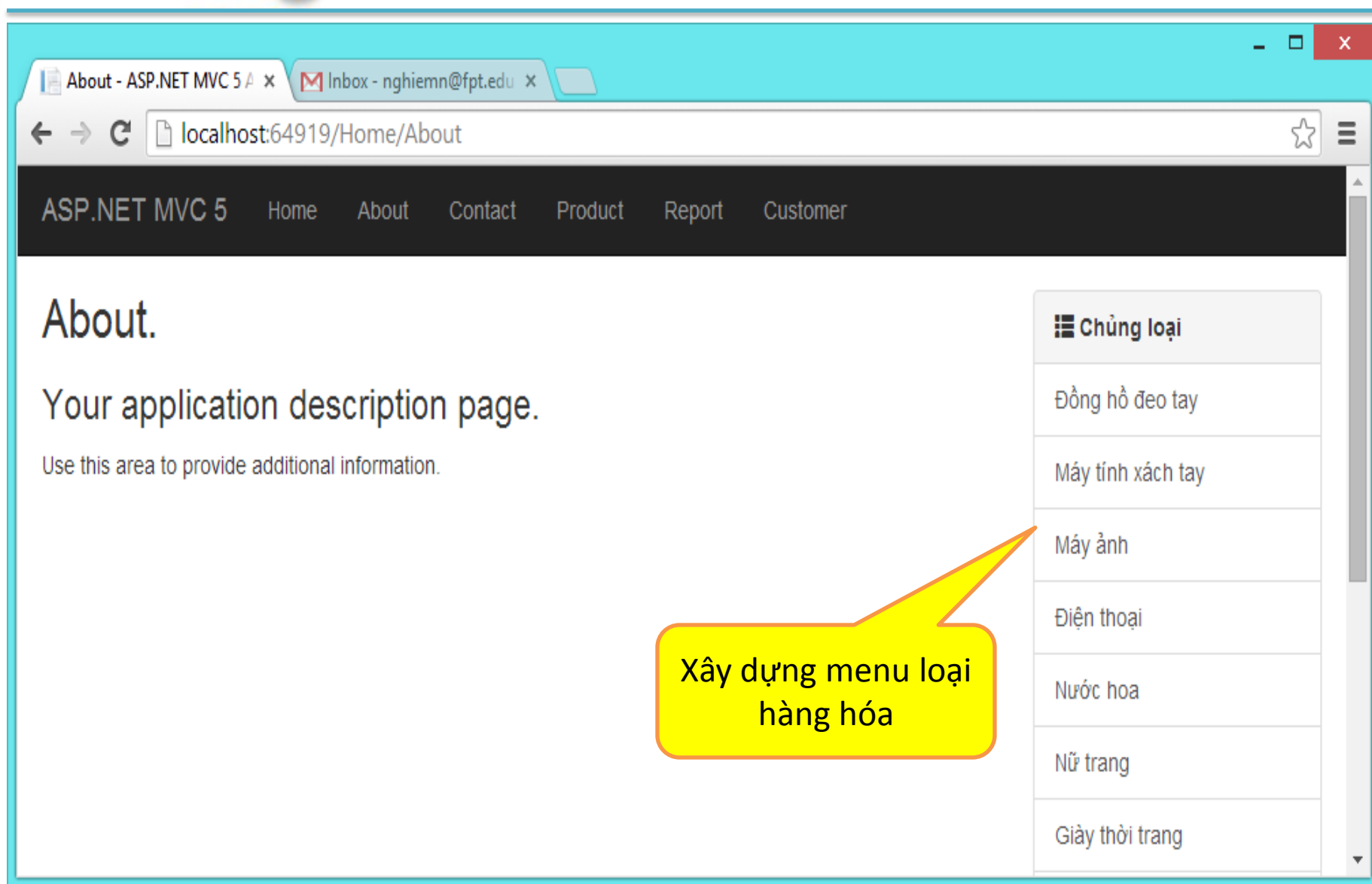
```
// GET:/Category/Edit/123
public ActionResult Edit(int id)
{
    var model = db.Categories.Find(id);
    return View(model);
}

// POST:/Category/Edit
[HttpPost]
public ActionResult Edit(Category model)
{
    db.Entry(model).State = EntityState.Modified;
    db.SaveChanges();
    return View(model);
}
```

| Id   | Name    | NameVN            |   |
|------|---------|-------------------|---|
| 1000 | Watches | Đồng hồ đeo tay   | <a href="#">Edit</a> : <a href="#">Delete</a> |
| 1001 | Laptops | Máy tính xách tay | <a href="#">Edit</a> : <a href="#">Delete</a> |
| 1002 | Cameras | Máy ảnh           | <a href="#">Edit</a> : <a href="#">Delete</a> |

Name

NameVN



About - ASP.NET MVC 5 x    Inbox - ngghiemn@fpt.edu x

localhost:64919/Home/About

ASP.NET MVC 5   Home   About   Contact   Product   Report   Customer

## About.

Your application description page.

Use this area to provide additional information.

**Chủng loại**

- Đồng hồ đeo tay
- Máy tính xách tay
- Máy ảnh
- Điện thoại
- Nước hoa
- Nữ trang
- Giày thời trang







Xây dựng menu loại hàng hóa

Product List - ASP.NET MVC5

localhost:64919/Product/List?CategoryId=1000

ASP.NET MVC 5 Home About Contact Product Report Customer

**Xây dựng trang hàng hóa**


|  |  |   |   |
|--|--|---|---|
| <p>Aniseed Syrup</p>  <p>\$190.00</p> | <p>Change</p>  <p>\$19.00</p> | <p>Guaranaj Fantajstica</p>  <p>\$4.50</p> | <p><b>Chủng loại</b></p> <ul style="list-style-type: none"><li>Đồng hồ đeo tay</li><li>Máy tính xách tay</li><li>Máy ảnh</li><li>Điện thoại</li><li>Nước hoa</li><li>Nữ trang</li><li>Giày thời trang</li></ul> |
| <p>Sasquatch Ale</p>                | <p>Steeleye Stout</p>       | <p>Cate de Blaye</p>                     |   |

Detail - ASP.NET MVC 5 A x

localhost:64919/Product/Detail/1055

ASP.NET MVC 5 Home About Contact Product Report Customer

## Chi tiết hàng hóa



- Mã: 1055
- Tên: Pacta chinois
- Đơn giá: 24
- Ngày SX: 7/8/2007 12:00:00 AM
- Chủng loại: Nữ trang
- Trạng thái: Còn hàng

EmEditor uses JavaScript or VBScript for its macro language, so those who are familiar with HTML or Windows scripting will be able to create macros with little difficulty. For those unfamiliar with scripting, EmEditor can record keystrokes, which can then be saved in a macro file that can easily be loaded in various situations. With the use of JavaScript or VBScript, you can also troubleshoot your code easily. For example, with JavaScript, you can use the following statement to troubleshoot errors

### Hàng cùng loại

|  |                                    |                                 |
|--|------------------------------------|---------------------------------|
| 1. <a href="#">Pacta chinois</a>           | 2. <a href="#">Mishi Kobe Niku</a> | 3. <a href="#">Alice Mutton</a> |
| 4. <a href="#">Tharinger Rostbratwurst</a> | 5. <a href="#">Perth Pasties</a>   | 6. <a href="#">Tourtiare</a>    |
| 7. <a href="#">Mishi Kobe Niku</a>         |                                    |                                 |

### Chủng loại

- Đồng hồ đeo tay
- Máy tính xách tay
- Máy ảnh
- Nữ trang
- Giày thời trang
- Túi xách du lịch
- Loại mới

Xây dựng trang chi tiết hàng hóa

Hàng cùng loại với hàng hóa đang xem



Customer Management - x

localhost:64919/Customer/Edit/ALFKI

ASP.NET MVC 5 Home About Contact Product Report Customer

## Edit Customer

Mã đăng nhập:

Mật khẩu:

Họ và tên:

Địa chỉ email:

Hình ảnh:

Trạng thái kích hoạt: ☒

## Customer List

| Id    | Password | Fullname     | Email           | Photo     | Activated? |                      |
|-------|----------|--------------|-----------------|-----------|------------|----------------------|
| ALFKI | iloveyou | Maria Anders | ALFKI@yahoo.com | ALFKI.jpg | Yes        | <a href="#">Edit</a> |
| ANATR | iloveyou | Ana Truiillo | ANATR@gmail.com | ANATR.jpg | No         | <a href="#">Edit</a> |

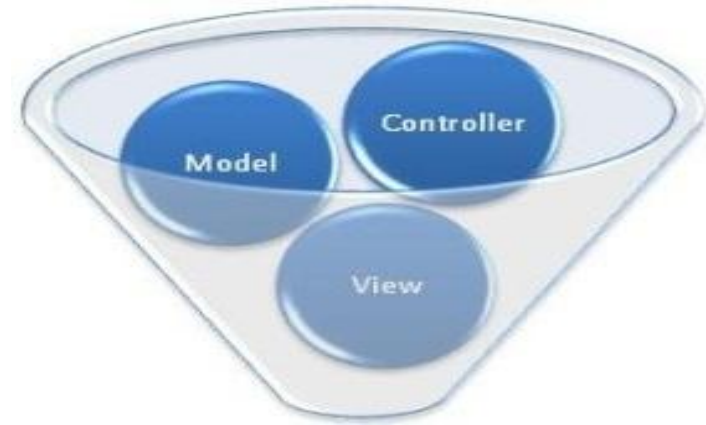
**Chủng loại**

- Đồng hồ đeo tay
- Máy tính xách tay
- Máy ảnh
- Điện thoại
- Nước hoa
- Nữ trang
- Giày thời trang
- Túi xách du lịch
- Loại mới
- Thời trang
- Thời trang

Xây dựng trang quản lý khách hàng

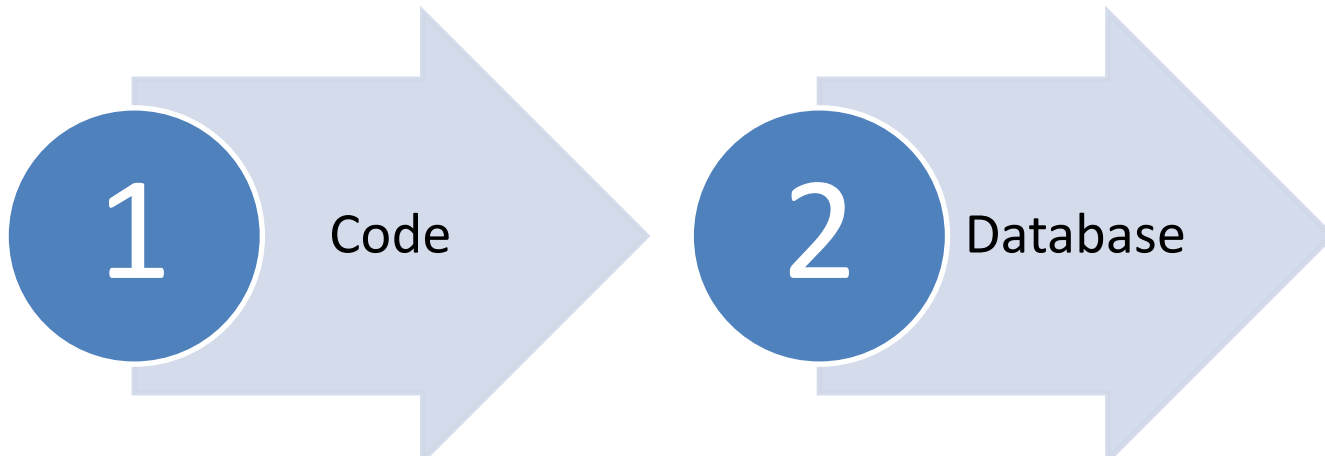
Cho phép upload hình lên server

Microsoft  
**ASP.net MVC 5**

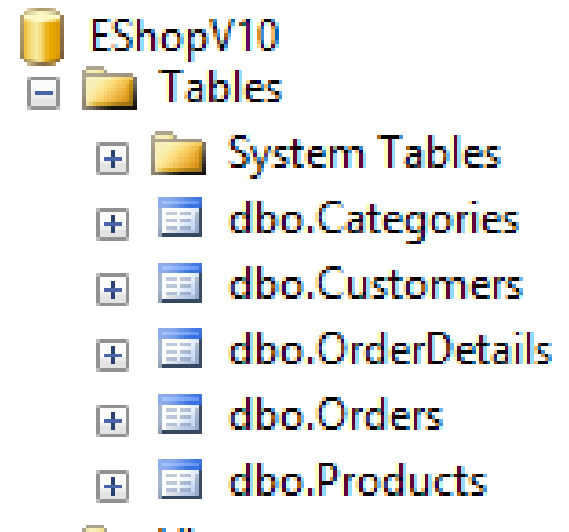


**CODE FIRST**  
CODE FIRST

ThS. Nguyễn Nghiệm  
0913.745.789  
nghiemn@fpt.edu.vn  
songlong2k@gmail.com



- ☐ Entity
  - ✓ Category
  - ✓ Product
  - ✓ Customer
  - ✓ Order
  - ✓ OrderDetail
- ☐ Database Context
  - ✓ EShopDbContext
  - ✓ Web.config

- 
- EShopV10
- [-] Tables
    - [+] System Tables
    - [+] dbo.Categories
    - [+] dbo.Customers
    - [+] dbo.OrderDetails
    - [+] dbo.Orders
    - [+] dbo.Products
    - ...

```
public class Product
{
    public int Id { get; set; }
    public string Name { get; set; }
    public double UnitPrice { get; set; }
    public string Image { get; set; }
    public DateTime ProductDate { get; set; }
    public bool Available { get; set; }
    public int CategoryId { get; set; }
    public string Description { get; set; }
}
```

```
public class OrderDetail
{
    public int Id { get; set; }
    public int OrderId { get; set; }
    public int ProductId { get; set; }
    public double UnitPrice { get; set; }
    public int Quantity { get; set; }
}
```

```
public class Category
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string NameVN { get; set; }
    public virtual List<Product> Products { get; set; }
}
```

```
public class Customer
{
    public string Id { get; set; }
    public string Password { get; set; }
    public string FullName { get; set; }
    public string Email { get; set; }
    public string Photo { get; set; }
    public bool Activated { get; set; }
    public virtual List<Order> Orders { get; set; }
}
```

```
public class Order
{
    public int Id { get; set; }
    public string CustomerId { get; set; }
    public DateTime OrderDate { get; set; }
    public string Address { get; set; }
    public double Amount { get; set; }
    public string Description { get; set; }
}
```

```
public virtual Customer Customer { get; set; }
public virtual List<OrderDetail> OrderDetails { get; set; }
```

Thực thể kết hợp: Nhiều-1 và 1-Nhiều

```
public class EShopV10 : DbContext
{
    public EShopV10() : base("name=EShopV10") { }

    public DbSet<Category> Categories { get; set; }
    public DbSet<Customer> Customers { get; set; }
    public DbSet<OrderDetail> OrderDetails { get; set; }
    public DbSet<Order> Orders { get; set; }
    public DbSet<Product> Products { get; set; }
}
```

```
<connectionStrings>
    <add name="EShopV10"
        connectionString="Server=.;Database=EShopV10;Integrated Security=True"
        providerName="System.Data.SqlClient" />
</connectionStrings>
```



```
[Table("KhoaHoc")]
public class Course
{
    [Key]
    [Column("MaKH")]
    public int Id { get; set; }
    [Column("TenKH")]
    public String Name { get; set; }
    [Column("HocPhi")]
    public double UnitPrice { get; set; }
}
```

- ❑ Tên thực thể **số ít** sẽ ánh xạ với bảng cùng tên **số nhiều**.
  - ✎ Tùy biến với `[Table("<tên bảng>")]`
- ❑ Tên thuộc tính **cùng tên** với tên cột.
  - ✎ Tùy biến với `[Column("<tên cột>")]`

- ❑ Tên thuộc tính khóa phải là **Id** hoặc **EntityId**.

✎ Tùy biến với `[Key]`

- ❑ **Khóa int** được hiểu là **tự tăng**.

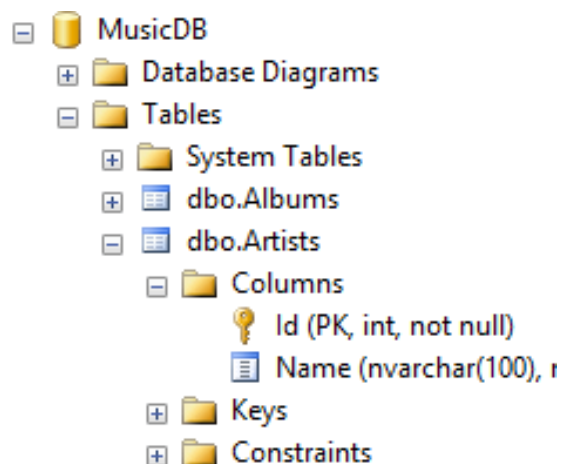
✎ Tùy biến với `[DatabaseGenerated(DatabaseGeneratedOption.Identity)]`

```
public class MvcApplication : System.Web.HttpApplication
{
    protected void Application_Start()
    {
        Database.SetInitializer(new MusicStoreDbInitializer());
    }
}
```



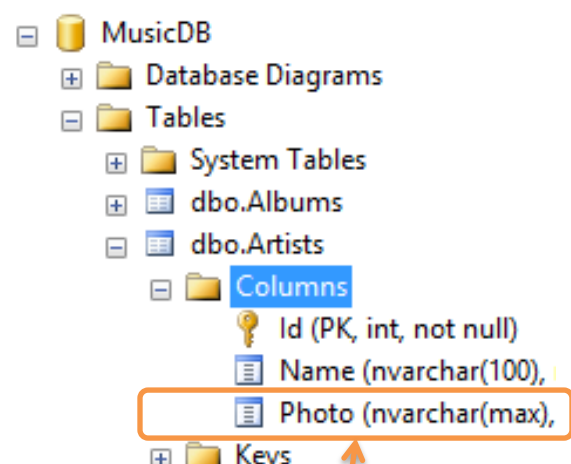
```
public class MusicStoreDbInitializer
    : DropCreateDatabaseIfModelChanges<MusicStoreDbContext>
{
    protected override void Seed(MusicStoreDbContext context)
    {
        base.Seed(context);
        var genres = new List<Genre>
        {
            new Genre { Name = "Rock" },
        };
        var artists = new List<Artist>
        {
            new Artist { Name = "Aaron Copland & London Symphony Orchestra" },
        };
        var albums = new List<Album>
        {
            new Album { Title = "A Copland Celebration, Vol. I", Genre =
            };
        };
        albums.ForEach(a => context.Albums.Add(a));
    }
}
```

- Chỉ cần 1 sự thay đổi các thuộc tính của EntityClass thì CSDL sẽ bị xóa đi và tạo lại



```
public class Artist
{
    public int Id { get; set; }
    [Required]
    [StringLength(100)]
    public String Name { get; set; }

    public List<Album> Albums { get; set; }
}
```

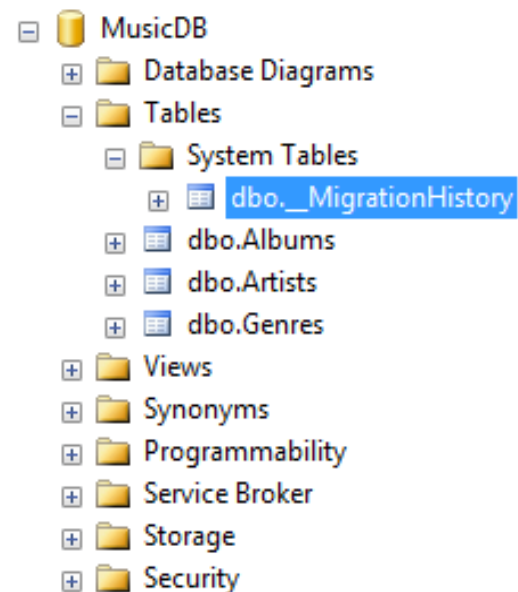


```
public class Artist
{
    public int Id { get; set; }
    [Required]
    [StringLength(100)]
    public String Name { get; set; }
    public String Photo { get; set; }

    public List<Album> Albums { get; set; }
}
```

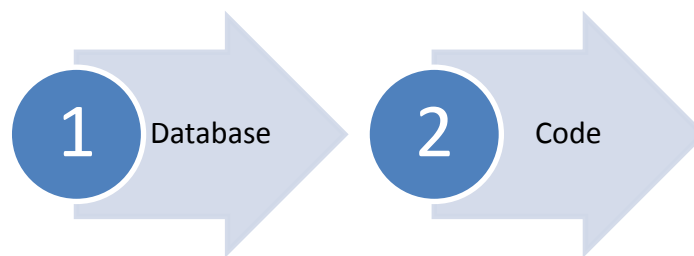
- ❑ Trong CSDL được sinh ra bởi mô hình lập trình Code-First của EF có chứa 1 bảng có tên là **\_MigrationHistory** được sử dụng để theo dõi phiên bản ánh xạ.

- ❑ Xóa bảng này sẽ làm mất liên lạc về sự thay đổi của EntityClass khi đó sẽ trở lại làm việc như CSDL đã tồn tại.



❑ Nếu CSDL của bạn đã có, thì bạn vẫn thực hiện các bước như cũ

- ~~✍~~ Định nghĩa Entity Class
- ~~✍~~ Định nghĩa DbContext Class
- ~~✍~~ Khai báo Connection String



❑ Ngoại trừ khai báo khởi đầu giá trị cho các bảng trong CSDL tại Global.asax

```
public class MvcApplication : System.Web.HttpApplication
{
    protected void Application_Start()
    {
        Database.SetInitializer(new MusicStoreDbInitializer());
    }
}
```

Bỏ dòng mã lệnh này