

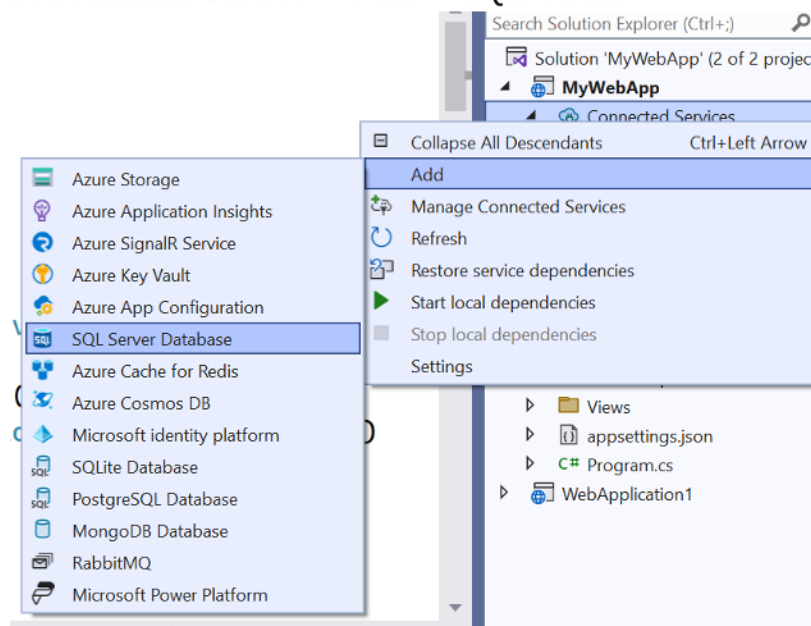
BÀI THỰC HÀNH 4

Data Access – Truy cập CSDL bằng Entity Framework

- **Code First: Từ Model => Sinh ra Database**
- **Database First: Từ Database => Sinh ra Model**

Hướng dẫn

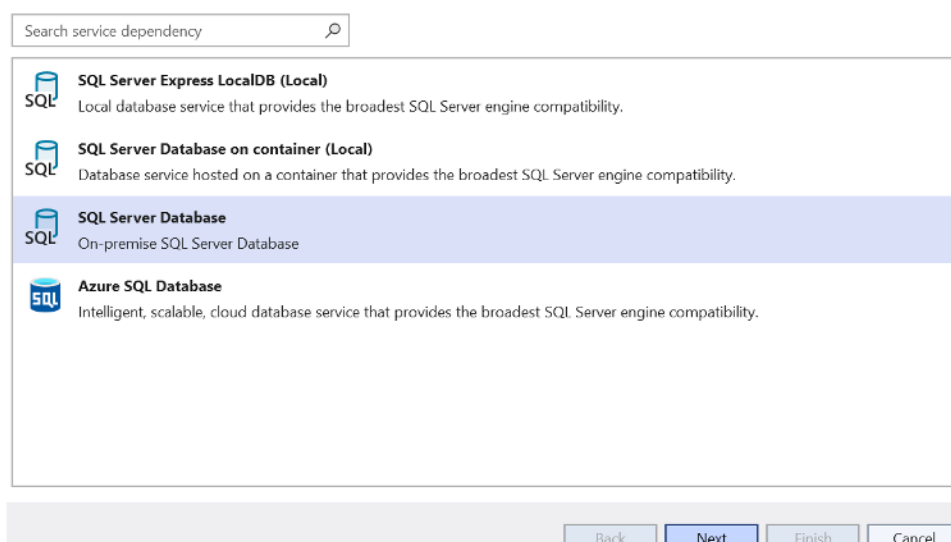
1. Cấu hình
 - Bật chế độ SQL Authentication
 - Enable account: System Administrator (SA)
 - Restart Server
2. Tạo Connection String – chuỗi kết nối tới cơ sở dữ liệu SQL Server
 - Right click Connected Service -> Add -> SQL Server



- Chọn SQLServer Database

Connect to dependency

Select a service dependency to add to your application.



- Đặt tên cho Connection String là: SchoolContext

Connect to SQL Server Database

Provide connection string and specify how to save it

Connection string name

Connection string value

Save connection string value in [Learn more](#)

☒ Local user secrets file: **Secrets.json (Local)**

☐ Azure Key Vault

☐ Do not save value anywhere

- Click ... trong Connection String Value, khai báo tên Server, tài khoản (sa, password)

SQL Connection String ? X

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:

Server name:

Log on to the server

Authentication:

User name:

Password:

☐ Save my password

Connect to a database

☒ Select or enter a database name:

☐ Attach a database file:

Logical name:

- Click OK để lưu Connection String vào file Secrets.json

3. Cài đặt EF: Có 2 cách

- #### 3.1. Chế độ dòng lệnh: Tools/NuGet Package Manager/Package Manager Console (PMC)

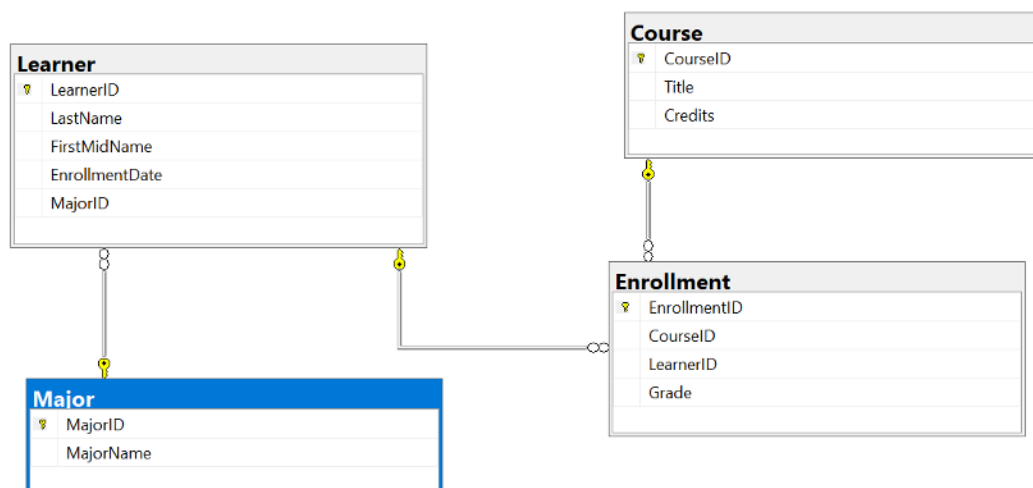
`Install-Package Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore -v 6.0.10`

`Install-Package Microsoft.EntityFrameworkCore.SqlServer -v 6.0.10`

- #### 3.2. Chế độ dòng lệnh: Tools/NuGet Package Manager/Manage Packages For Solution

Lần lượt tìm 2 gói trên, chọn phiên bản 6.0.10 và cài đặt

4. Tạo các model cho cơ sở dữ liệu



4.1. Major.cs

Trong thư mục Model, tạo file Major.cs để định nghĩa lớp Model Major gồm các thuộc tính sau:

```
namespace MyWebApp.Models
{
    public class Major
    {
        public Major() {
            Learners = new HashSet<Learner>();
        }
        public int MajorID { get; set; }
        public string MajorName { get; set; }
        public virtual ICollection<Learner> Learners { get; set; }
    }
}
```

4.2. Learner.cs

```
namespace MyWebApp.Models
{
    public class Learner
    {
        public Learner()
        {
            Enrollments = new HashSet<Enrollment>();
        }
        public int LearnerID { get; set; }
        public string LastName { get; set; }
        public string FirstMidName { get; set; }
        public DateTime EnrollmentDate { get; set; }
        public int MajorID { get; set; }
        public virtual Major? Major { get; set; }
        public virtual ICollection<Enrollment> Enrollments { get; set; }
    }
}
```

4.3 Course.cs

```
using MyWebApp.Models;

namespace MyWebApp.Models
{
    public class Course
    {
        public Course() {
            Enrollments = new HashSet<Enrollment>();
        }
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int CourseID { get; set; }
        public string Title { get; set; }
        public int Credits { get; set; }
        public virtual ICollection<Enrollment> Enrollments { get; set; }
    }
}
```

4.4. Enrollment.cs

```
namespace MyWebApp.Models
{
    public class Enrollment
    {
        public int EnrollmentID { get; set; }
        public int CourseID { get; set; }
        public int LearnerID { get; set; }
        public float Grade { get; set; }
        public virtual Learner? Learner { get; set; }
        public virtual Course? Course { get; set; }
    }
}
```

5. Tạo thư mục Data, trong đó

5.1. Tạo lớp SchoolContext kế thừa từ DbContext mô tả toàn bộ cơ sở dữ liệu

```
using Microsoft.EntityFrameworkCore;
using MyWebApp.Models;

namespace MyWebApp.Data
{
    public class SchoolContext : DbContext
    {
        public SchoolContext(DbContextOptions<SchoolContext> options)
            : base(options) { }
        public virtual DbSet<Course> Courses { get; set; }
        public virtual DbSet<Learner> Learners { get; set; }
        public virtual DbSet<Enrollment> Enrollments { get; set; }
        public virtual DbSet<Major> Majors { get; set; }
        protected override void OnModelCreating(ModelBuilder modelBuilder)
        {
            modelBuilder.Entity<Major>().ToTable(nameof(Major));
            modelBuilder.Entity<Course>().ToTable(nameof(Course));
            modelBuilder.Entity<Learner>().ToTable(nameof(Learner));
            modelBuilder.Entity<Enrollment>().ToTable(nameof(Enrollment));
        }
    }
}
```

5.2. Tạo dữ liệu mẫu: Tạo lớp DbInitializer để khởi tạo dữ liệu

```
using Microsoft.EntityFrameworkCore;
using MyWebApp.Models;
namespace MyWebApp.Data
{
    public class DbInitializer
    {
        public static void Initialize(IServiceProvider serviceProvider)
        {
            using(var context = new SchoolContext(serviceProvider
                .GetRequiredService<DbContextOptions<SchoolContext>>()))
            {
                context.Database.EnsureCreated();
                if (context.Majors.Any()){
                    return;
                }
                var majors = new Major[] {
                    new Major{MajorName="IT"},
                    new Major{MajorName="Economics"},
                    new Major{MajorName="Mathematics"},
                };
                foreach(var major in majors){
                    context.Majors.Add(major);
                }
                context.SaveChanges();
                var learners = new
                Learner[] {
                    new Learner { FirstMidName = "Carson", LastName = "Alexander",
                        EnrollmentDate = DateTime.Parse("2005-09-01") , MajorID = 1},
                    new Learner { FirstMidName = "Meredith", LastName = "Alonso",
                        EnrollmentDate = DateTime.Parse("2002-09-01"), MajorID = 2 }
                };
                foreach (Learner l in learners){
                    context.Learners.Add(l);
                }
                context.SaveChanges();
                var courses = new Course[]{
                    new Course{CourseID=1050,Title="Chemistry",Credits=3},
                    new Course{CourseID=4022,Title="Microeconomics",Credits=3},
                    new Course{CourseID=4041,Title="Macroeconomics",Credits=3}
                };
                foreach (Course c in courses){
                    context.Courses.Add(c);
                }
                context.SaveChanges();
                var enrollments = new Enrollment[]{
                    new Enrollment{LearnerID=1,CourseID=1050,Grade=5.5f},
                    new Enrollment{LearnerID=1,CourseID=4022,Grade=7.5f},
                    new Enrollment{LearnerID=2,CourseID=1050,Grade=3.5f},
                    new Enrollment{LearnerID=2,CourseID=4041,Grade=7f}
                };
                foreach (Enrollment e in enrollments){
                    context.Enrollments.Add(e);
                }
                context.SaveChanges();
            }
        }
    }
}
```

6. Đăng ký SchoolContext và gọi khởi tạo dữ liệu mẫu trong Program.cs

```
using Microsoft.EntityFrameworkCore;
using MyWebApp.Data;
using MyWebApp.Models;

var builder = WebApplication.CreateBuilder(args);
//builder.Configuration
// .SetBasePath(Directory.GetCurrentDirectory()).AddJsonFile("Secrets.json");
//Đăng ký SchoolContext là một DbContext của ứng dụng
builder.Services.AddDbContext<SchoolContext>(options => options
.UseSqlServer(builder.Configuration.GetConnectionString("SchoolContext")));

// Add services to the container.
builder.Services.AddControllersWithViews();

var app = builder.Build();

using(var scope = app.Services.CreateScope())
{
    var services = scope.ServiceProvider;
    DbInitializer.Initialize(services);
}

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
    // The default HSTS value is 30 days. You may want to change this for production
    scenarios, see https://aka.ms/aspnetcore-hsts.
    app.UseHsts();
}

app.UseHttpsRedirection();
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");

app.Run();
```

7. Run dự án và kiểm tra dữ liệu trong cơ sở dữ liệu SQL Server.

8. Khai thác cơ sở dữ liệu: Xây dựng chức năng CRUD trên bảng Learner

8.1 List (R) – hiển thị danh sách các Learner

8.1.1 Tạo LearnerController với Action Index được viết như sau

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Rendering;
using Microsoft.EntityFrameworkCore;
using MyWebApp.Data;
using MyWebApp.Models;

namespace MyWebApp.Controllers
{
    public class LearnerController : Controller
    {
        private SchoolContext db;
        public LearnerController(SchoolContext context)
        {
            db = context;
        }
        public IActionResult Index()
        {
            var learners = db.Learners.Include(m => m.Major).ToList();
            return View(learners);
        }
    }
}
```

8.1.2. Tạo View Index.cshtml cho Action Index như sau

```
@{
    ViewData["Title"] = "Index";
    Layout = "~/Views/Shared/MyLayout.cshtml";
}
@model IEnumerable<Learner>
<div class="ms-2 mx-2">
    <h1>Learner List</h1>
    <table class="table table-striped">
        <thead>
            <tr>
                <th>Mã</th>
                <th>Tên</th>
                <th>Họ</th>
                <th>Ngành</th>
                <th>Ngày Nhập Học</th>
                <th>Chức năng</th>
            </tr>
        </thead>
        <tbody>
            @foreach (var s in Model)
            {
                <tr>
                    <td>@s.LearnerID</td>
                    <td>@s.FirstMidName</td>
                    <td>@s.LastName</td>
                    <td>@s.Major.MajorName</td>
                    <td>@s.EnrollmentDate</td>
                    <td><a asp-action="Edit" asp-route-id="@s.LearnerID">Edit</a> | <a asp-
action="Delete" asp-route-id="@s.LearnerID">Delete</a></td>
                </tr>
            }
        </tbody>
    </table>
    <a asp-action="Create" class="btn btn-primary">Create New Student</a>
</div>
```


8.1.3 Kết quả chạy Learner/Index để hiển thị danh sách người học

Mã	Tên	Họ	Ngành	Ngày Nhập Học	Chức năng
1	Carson	Alexander	IT	9/1/2005 12:00:00 AM	Edit / Delete
2	Meredith	Alonso	Economics	9/1/2002 12:00:00 AM	Edit / Delete
5	Nam	Lê	Mathematics	10/4/2023 1:59:00 PM	Edit / Delete

8.2 Create (C) – Thêm một Learner

8.2.1. Trong LearnerController, tạo thêm 2 action Create, một cho phương thức get, một cho phương thức post

```
namespace MyWebApp.Controllers
{
    public class LearnerController : Controller
    {
        //thêm 2 action create
        public IActionResult Create()
        {
            //dùng 1 trong 2 cách để tạo SelectList gửi về View qua ViewBag để
            //hiển thị danh sách chuyên ngành (Majors)
            var majors = new List<SelectListItem>(); //cách 1
            foreach (var item in db.Majors)
            {
                majors.Add(new SelectListItem { Text = item.MajorName,
                    Value = item.MajorID.ToString() });
            }
            ViewBag.MajorID = majors;
            ViewBag.MajorID = new SelectList(db.Majors, "MajorID", "MajorName"); //cách 2
            return View();
        }

        [HttpPost]
        [ValidateAntiForgeryToken]
        public IActionResult Create([Bind("FirstMidName, LastName, MajorID, EnrollmentDate")]
Learner learner)
        {
            if (ModelState.IsValid)
            {
                db.Learners.Add(learner);
                db.SaveChanges();
                return RedirectToAction(nameof(Index));
            }
            //lại dùng 1 trong 2 cách tạo SelectList gửi về View để hiển thị danh sách Majors
            ViewBag.MajorID = new SelectList(db.Majors, "MajorID", "MajorName");
            return View();
        }
    }
}
```

8.2.2. Tạo View Create.cshtml

```
@{
    ViewData["Title"] = "Create";
    Layout = "~/Views/Shared/MyLayout.cshtml";
}
@model Learner
<div class="ms-2 mx-2">
    <h1>Create</h1>
    <div asp-validation-summary="All" class="text-danger"></div>
    <form asp-controller="Learner" asp-action="Create" method="post" class="mt-3">
        <div style="margin-top:7px" class="form-group row">
            <label asp-for="FirstMidName" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="FirstMidName" class="form-control" placeholder="First Mid Name">
            </div>
            <span asp-validation-for="FirstMidName" class="text-danger"></span>
        </div>
        <div style="margin-top:7px" class="form-group row">
            <label asp-for="LastName" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="LastName" class="form-control" placeholder="Last Name">
            </div>
            <span asp-validation-for="LastName" class="text-danger"></span>
        </div>

        <div style="margin-top:7px" class="form-group row">
            <label asp-for="MajorID" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <select asp-for="MajorID" class="custom-select mr-sm-2"
                    asp-items="@ViewBag.MajorID"></select>
            </div>
            <span asp-validation-for="MajorID" class="text-danger"></span>
        </div>

        <div style="margin-top:7px" class="form-group row">
            <label asp-for="EnrollmentDate" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="EnrollmentDate" class="form-control">
            </div>
            <span asp-validation-for="EnrollmentDate" class="text-danger"></span>
        </div>

        <div style="margin-top:10px" class="form-group row">
            <div class="col-sm-10">
                <button type="submit" class="btn btn-primary">Create</button>
            </div>
        </div>
    </form>
</div>
```

8.2.3. Kết quả chạy ứng dụng

Trong Learner/Index click vào nút “Create New Student” sẽ mở ra màn hình để thêm mới một người học

Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Create

FirstMidName: Minh

LastName: Trần

MajorID: Economics

EnrollmentDate: 10/04/2023 02:47 PM

Copyright © 2023

Logged in as: MyAccount

Nhập đầy đủ các thông tin của người học và click vào nút Create sẽ tạo một người học mới, thêm vào csdl và hiển thị lại danh sách đã cập nhật người học mới

Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Create New Student

Learner List

Mã	Tên	Họ	Ngành	Ngày Nhập Học	Chức năng
1	Carson	Alexander	IT	9/1/2005 12:00:00 AM	Edit Delete
2	Meredith	Alonso	Economics	9/1/2002 12:00:00 AM	Edit Delete
5	Nam	Lê	Mathematics	10/4/2023 1:59:00 PM	Edit Delete
6	Hà	Nguyễn	Economics	10/4/2023 2:46:00 PM	Edit Delete
7	Minh	Trần	Economics	10/4/2023 2:47:00 PM	Edit Delete

Copyright © 2023

Logged in as: MyAccount

8.3 Update (U) – Sửa một Learner

8.3.1. Trong `LearnerController`, tạo thêm 2 action `Edit`, một cho phương thức `get`, một cho phương thức `post`

```
namespace MyWebApp.Controllers
{
    public class LearnerController : Controller
    {
        //thêm 2 action edit
        public IActionResult Edit(int id)
        {
            if (id == null || db.Learners == null)
            {
                return NotFound();
            }

            var learner = db.Learners.Find(id);
            if (learner == null)
            {
                return NotFound();
            }
            ViewBag.MajorId = new SelectList(db.Majors, "MajorID", "MajorName", learner.MajorID);
            return View(learner);
        }

        [HttpPost]
        [ValidateAntiForgeryToken]
        public IActionResult Edit(int id,
            [Bind("LearnerID,FirstMidName,LastName,MajorID,EnrollmentDate")] Learner learner)
        {
            if (id != learner.LearnerID)
            {
                return NotFound();
            }

            if (ModelState.IsValid)
            {
                try
                {
                    db.Update(learner);
                    db.SaveChanges();
                }
                catch (DbUpdateConcurrencyException)
                {
                    if (!LearnerExists(learner.LearnerID))
                    {
                        return NotFound();
                    }
                    else
                    {
                        throw;
                    }
                }
                return RedirectToAction(nameof(Index));
            }
            ViewBag.MajorId = new SelectList(db.Majors, "MajorID", "MajorName", learner.MajorID);
            return View(learner);
        }
        private bool LearnerExists(int id)
        {
            return (db.Learners?.Any(e => e.LearnerID == id)).GetValueOrDefault();
        }
    }
}
```

8.3.2 Tạo View Edit.cshtml

```
@{
    ViewData["Title"] = "Edit";
    Layout = "~/Views/Shared/MyLayout.cshtml";
}
@model Learner
<div class="ms-2 mx-2">
    <h1>Edit</h1>
    <div asp-validation-summary="All" class="text-danger"></div>
    <form asp-controller="Learner" asp-action="Edit" asp-route-id="@Model.LearnerID"
method="post" class="mt-3">
        <input type="hidden" asp-for="LearnerID"/>
        <div style="margin-top:7px" class="form-group row">
            <label asp-for="FirstMidName" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="FirstMidName" class="form-control" placeholder="First Mid Name">
            </div>
            <span asp-validation-for="FirstMidName" class="text-danger"></span>
        </div>
        <div style="margin-top:7px" class="form-group row">
            <label asp-for="LastName" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="LastName" class="form-control" placeholder="Last Name">
            </div>
            <span asp-validation-for="LastName" class="text-danger"></span>
        </div>

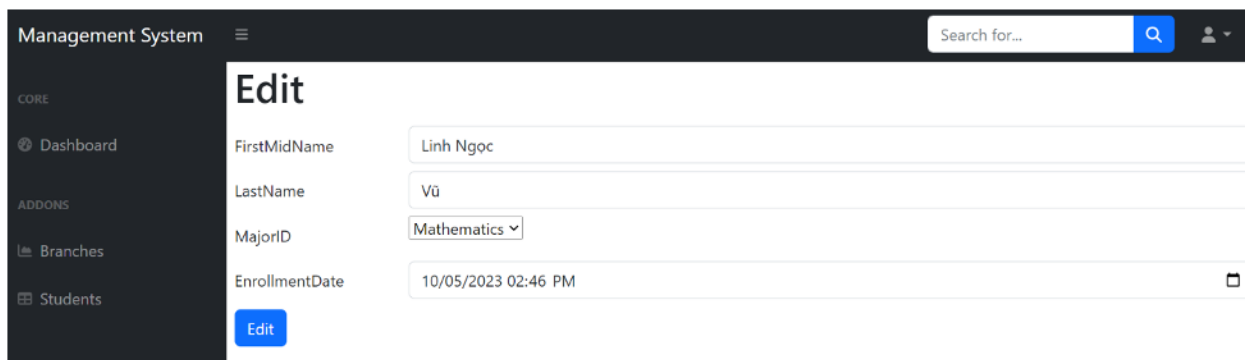
        <div style="margin-top:7px" class="form-group row">
            <label asp-for="MajorID" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <select asp-for="MajorID" class="custom-select mr-sm-2"
                    asp-items="@ViewBag.MajorID"></select>
            </div>
            <span asp-validation-for="MajorID" class="text-danger"></span>
        </div>

        <div style="margin-top:7px" class="form-group row">
            <label asp-for="EnrollmentDate" class="col-sm-2 col-form-label"></label>
            <div class="col-sm-10">
                <input asp-for="EnrollmentDate" class="form-control">
            </div>
            <span asp-validation-for="EnrollmentDate" class="text-danger"></span>
        </div>

        <div style="margin-top:10px" class="form-group row">
            <div class="col-sm-10">
                <button type="submit" class="btn btn-primary">Edit</button>
            </div>
        </div>
    </form>
</div>
```

8.3.3. Kết quả chạy ứng dụng

Trong Learner/Index click vào link “Edit” sẽ mở ra màn hình hiển thị thông tin cũ của người học để sửa đổi



Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Edit

FirstMidName: Linh Ngoc

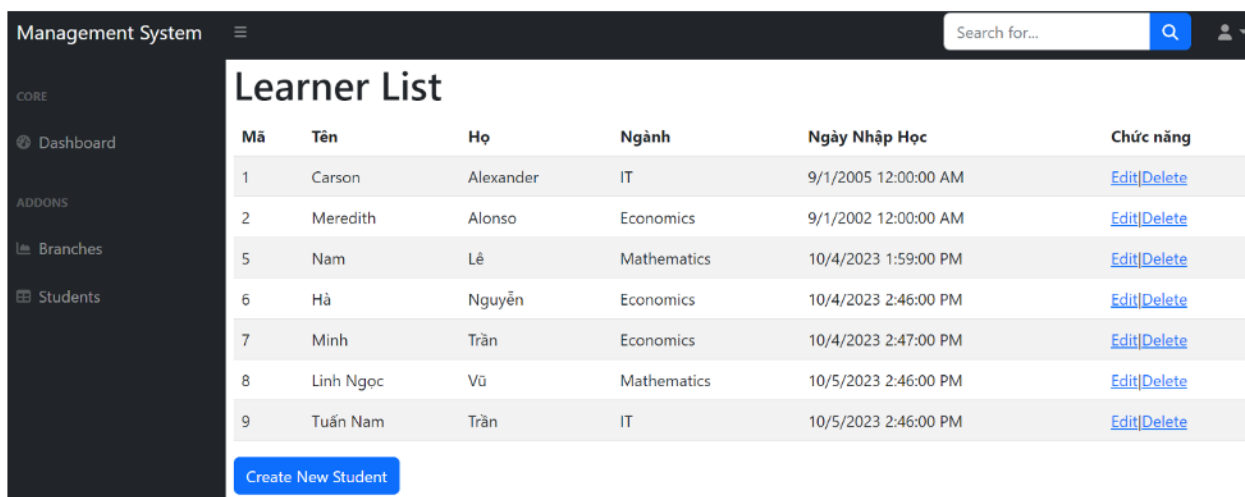
LastName: Vũ

MajorID: Mathematics

EnrollmentDate: 10/05/2023 02:46 PM

Edit

Sửa đổi thông tin và click vào Edit sẽ lưu thông tin sau khi thay đổi trong cơ sở dữ liệu



Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Learner List

Mã	Tên	Họ	Ngành	Ngày Nhập Học	Chức năng
1	Carson	Alexander	IT	9/1/2005 12:00:00 AM	Edit Delete
2	Meredith	Alonso	Economics	9/1/2002 12:00:00 AM	Edit Delete
5	Nam	Lê	Mathematics	10/4/2023 1:59:00 PM	Edit Delete
6	Hà	Nguyễn	Economics	10/4/2023 2:46:00 PM	Edit Delete
7	Minh	Trần	Economics	10/4/2023 2:47:00 PM	Edit Delete
8	Linh Ngoc	Vũ	Mathematics	10/5/2023 2:46:00 PM	Edit Delete
9	Tuấn Nam	Trần	IT	10/5/2023 2:46:00 PM	Edit Delete

Create New Student

8.4 Delete (D) – Xóa một Learner

8.4.1. Trong LearnerController, tạo thêm 2 action Delete, một cho phương thức get, một cho phương thức post

```
namespace MyWebApp.Controllers
{
    public class LearnerController : Controller
    {
        //thêm 2 action edit
        public IActionResult Delete(int id)
        {
            if (id == null || db.Learners == null)
            {
                return NotFound();
            }

            var learner = db.Learners.Include(l => l.Major)
                .Include(e => e.Enrollments)
                .FirstOrDefault(m => m.LearnerID == id);
            if (learner == null)
            {
                return NotFound();
            }
            if(learner.Enrollments.Count()>0)
            {
                return Content("This learner has some enrollments, can't delete!");
            }
            return View(learner);
        }

        // POST: Learner/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public IActionResult DeleteConfirmed(int id)
        {
            if (db.Learners == null)
            {
                return Problem("Entity set 'Learners' is null.");
            }
            var learner = db.Learners.Find(id);
            if (learner != null)
            {
                db.Learners.Remove(learner);
            }

            db.SaveChanges();
            return RedirectToAction(nameof(Index));
        }
    }
}
```

8.4.2 Tạo View Delete.cshtml để hỏi lại người dùng “chắc chắn muốn xóa?”

```
@{
    ViewData["Title"] = "Delete";
    Layout = "~/Views/Shared/MyLayout.cshtml";
}
@model Learner

<div class="ms-2 mx-2">
    <h1>Delete</h1>
    <h3>Are you sure you want to delete this?</h3>
    <h4>Learner</h4>
    <hr />
    <dl class="row">
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.FirstMidName)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.FirstMidName)
        </dd>
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.LastName)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.LastName)
        </dd>
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.Major)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.Major.MajorName)
        </dd>
        <dt class="col-sm-2">
            @Html.DisplayNameFor(model => model.EnrollmentDate)
        </dt>
        <dd class="col-sm-10">
            @Html.DisplayFor(model => model.EnrollmentDate)
        </dd>
    </dl>

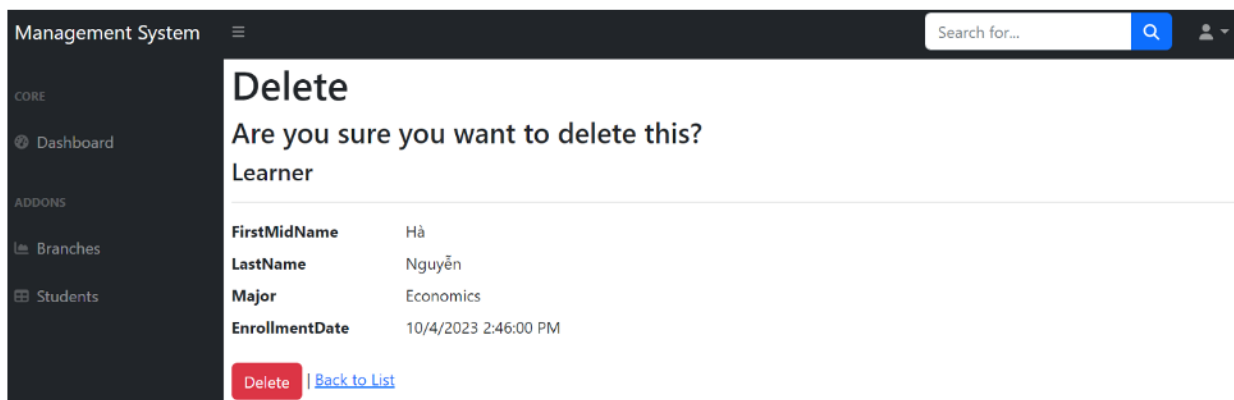
    <form asp-action="Delete">
        <input type="hidden" asp-for="@Model.LearnerID" />
        <input type="submit" value="Delete" class="btn btn-danger" /> |
        <a asp-action="Index">Back to List</a>
    </form>
</div>
```


8.3.3. Kết quả chạy ứng dụng

Trong Learner/Index click vào link “Delete”

Có ràng buộc dữ liệu, nếu người học hiện tại vẫn còn các bản ghi đăng ký học trong bảng Enrollment thì sẽ hiển thị thông báo không thể xóa người học này

Ngược lại, sẽ mở ra màn hình hiển thị thông tin của người học và hỏi lại người dùng có chắc chắn muốn xóa



Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Delete

Are you sure you want to delete this?

Learner

FirstMidName Hà

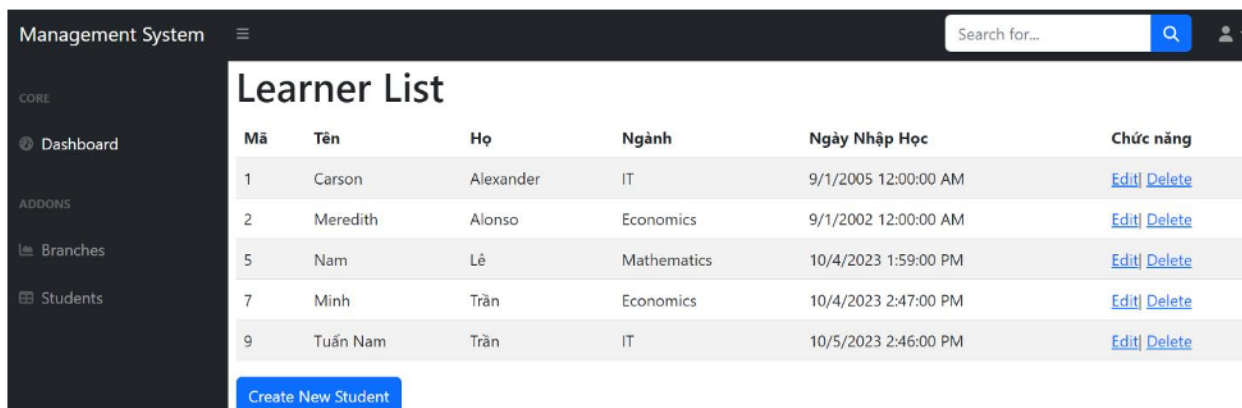
LastName Nguyễn

Major Economics

EnrollmentDate 10/4/2023 2:46:00 PM

[Delete](#) | [Back to List](#)

Kích vào nút Delete sẽ xóa người học



Management System

Search for...

CORE

Dashboard

ADDONS

Branches

Students

Learner List

Mã	Tên	Họ	Ngành	Ngày Nhập Học	Chức năng
1	Carson	Alexander	IT	9/1/2005 12:00:00 AM	Edit Delete
2	Meredith	Alonso	Economics	9/1/2002 12:00:00 AM	Edit Delete
5	Nam	Lê	Mathematics	10/4/2023 1:59:00 PM	Edit Delete
7	Minh	Trần	Economics	10/4/2023 2:47:00 PM	Edit Delete
9	Tuấn Nam	Trần	IT	10/5/2023 2:46:00 PM	Edit Delete

[Create New Student](#)