

Автор: Абдуллін Олексій

Група: КІТ-119а

Дата: 17.12.2021

## Лабораторна робота № 8

Тема: Основи ASP.NET MVC

Задачі:

1. Створити веб-застосунок за допомогою ASP.NET MVC.
2. Реалізувати додавання студенту у список, його видалення та редагування. Використовувати валідацію даних при їх введенні.

### Опис класів

Student – модель студента;

HomeController – клас контролер, що відповідає за роботу із класом Student;

### Текст програми

#### Student

```
namespace Abdullin08.Models
{
    public class Student
    {
        public int Id { get; set; }
        public string FirstName { get; set; }
        public string SurName { get; set; }
        public string Group { get; set; }
        public int AcademicPerformance { get; set; }
    }
}
```

#### ApplicationContext

```
using Microsoft.EntityFrameworkCore;

namespace Abdullin08.Data
{
    public class ApplicationContext : DbContext
    {
        public DbSet<Models.Student> Students { get; set; }

        public ApplicationContext(DbContextOptions<ApplicationContext> options)
            : base(options)
        { }
    }
}
```

```

        {
            Database.EnsureCreated();
        }
    }
}

```

## HomeController

```

using Abdullin08.Data;
using Abdullin08.Models;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.Logging;
using System.Diagnostics;
using System.Threading.Tasks;

namespace Abdullin08.Controllers
{
    public class HomeController : Controller
    {
        private readonly ILogger<HomeController> _logger;
        private ApplicationDbContext _db;
        public HomeController(ApplicationContext context)
        {
            _db = context;
        }
        public async Task<IActionResult> Index()
        {
            return View(await _db.Students.ToListAsync());
        }
        public IActionResult Create()
        {
            return View();
        }
        [HttpPost]
        public async Task<IActionResult> Create(Student student)
        {
            if (!Validation.isCorrect(student))
            {
                TempData["notice"] = "Error! Invalid data. Student was not registered.";
                return RedirectToAction("Index");
            }
            _db.Students.Add(student);
            await _db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        public async Task<IActionResult> Details(int? id)
        {
            if (id != null)
            {
                Student student = await _db.Students.FirstOrDefaultAsync(s => s.Id ==
id);

                if (student != null)
                    return View(student);
            }
            return NotFound();
        }
        public async Task<IActionResult> Edit(int? id)
        {
            if (id != null)
            {
                Student student = await _db.Students.FirstOrDefaultAsync(s => s.Id ==
id);

                if (student != null)
                    return View(student);
            }
        }
    }
}

```

```

        }
        return NotFound();
    }
    [HttpPost]
    public async Task<IActionResult> Edit(Student student)
    {
        _db.Students.Update(student);
        await _db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    [HttpGet]
    [ActionName("Delete")]
    public async Task<IActionResult> ConfirmDelete(int? id)
    {
        if (id != null)
        {
            Student student = await _db.Students.FirstOrDefaultAsync(s => s.Id ==
id);
            if (student != null)
                return View(student);
        }
        return NotFound();
    }

    [HttpPost]
    public async Task<IActionResult> Delete(int? id)
    {
        if (id != null)
        {
            Student student = await _db.Students.FirstOrDefaultAsync(s => s.Id ==
id);
            if (student != null)
            {
                _db.Students.Remove(student);
                await _db.SaveChangesAsync();
                return RedirectToAction("Index");
            }
        }
        return NotFound();
    }

    public IActionResult Privacy()
    {
        return View();
    }

    [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore =
true)]
    public IActionResult Error()
    {
        return View(new ErrorViewModel { RequestId = Activity.Current?.Id ??
HttpContext.TraceIdentifier });
    }
}

```

## Validation

```
using System.Text.RegularExpressions;

namespace Abdullin08.Models
{
    public static class Validation
    {
        public static bool isCorrect(Student student)
        {
            Regex regex_string = new Regex(@"^[a-z]+$", RegexOptions.IgnoreCase);
            Regex regex_group = new Regex(@"^[A-Z]{1,3}-[0-9]{2,3}[a-z]{1,2}$",
            RegexOptions.IgnoreCase);

            if (student.FirstName == null || !regex_string.IsMatch(student.FirstName))
            {
                return false;
            }
            else if (student.SurName == null || !regex_string.IsMatch(student.SurName) )
            {
                return false;
            }
            else if (student.Group == null || !regex_group.IsMatch(student.Group))
            {
                return false;
            }
            else if (student.AcademicPerformance == 0 || student.AcademicPerformance >
100 || student.AcademicPerformance < 0 )
            {
                return false;
            }
            return true;
        }
    }
}
```

## Результат роботи програми

Welcome			
Id	FirstName	SurName	Action
2	Oleksii	Abdullin	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
3	Volodymir	Vologda	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
4	Galina	Zolo	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
5	Drya	Kurgan	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
7	Galina	Zolo	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
<div>Add student</div>			

Рисунок 1 – Результати роботи програми

Sudent: Volodymir Vologda

Id	FirstName	SurName	Group	AcademicPerformance
3	Volodymir	Vologda	CIT-120a	75

Edit

Delete

Рисунок 2 – Details про студента

First Name

Volodymir

Surname

Vologda

Group

CIT-120a

Academic Performance

75

Submit

Рисунок 3 – Зміна інформації про студента

Sudent: Volodymir Vologda

Id	FirstName	SurName	Group	AcademicPerformance
3	Volodymir	Vologda	CIT-120a	89

Edit

Delete

Рисунок 4 – Змінена інформація про студента

Id	FirstName	SurName	Group	AcademicPerformance
3	Volodymir	Vologda	CIT-120a	89

Delete

Рисунок 5 – Видалення студента

Welcome			
Id	FirstName	SurName	Action
2	Oleksii	Abdullin	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
4	Galina	Zolo	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
5	Drya	Kurgan	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
7	Galina	Zolo	<a href="#">Details</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
<div>Add student</div>			

Рисунок 6 – Результат видалення студента

**Висновок:** У результаті виконання лабораторної роботи було створено веб-застосунок. У ньому було реалізовано додавання студенту у список, його видалення та редагування. Було використано валідацію даних при їх введенні.