Лабораторная работа 2. Разработка технического проекта на создание ИС средствами технологии объектного проектирования информационных систем *Часовских В.П., сайт http://vikchas.ru*

Разработка технического проекта на создание ИС основывается на технологии объектного проектирования ASP.NET Core MVC. Для начального освоения ASP.NET Core MVC предлагается реальный проект для подготовки и выполнения как показано в методической части. Предполагается, что студент знает как запустить Visual Studio 2017 Enterprise интегрированная комплексная платформа для групп любого размера с высокими требованиями к качеству и масштабу и полным набором инструментов и служб для разработки, создания сложных корпоративных приложений и управления ими.

Необходимо запустить Visual Studio 2017 Enterprise и последовательно выполнить последующие действия. После выполнения реального примера ИС можно приступить к разработке проекта.

Microsoft Visual Studio Φλέπ Προσκά Βνά Προσκά Κονισιμαο Οραρχία © •	ва Тест Инструменты R Анализ • Р Присоединение • ў	Окно Справка 5 7			T	Быстрый запуск (Ctrl+Q) Обсареватель решений	P = 5 Viktor Chas *	х 🖌 Уведом
37.00A	Создание проекта			7 X				ления
	эследние файлы	Сортировка: По умолчанию 🔹 🏥 📃		Поиск (Ctrl+E)				
ę.	тановленные	Be5-приложение ASP.NET Core	Visual C#	Тип: Visual C#				
v cipaninus cónorra SQL Server V	Asual C4 Wentergaansee repotocesser April V Ascaccescece englescense Windows Caccestance englescense Windows Cours Astronad Apple Mach Cours Apple V Apple Wach Cours Cours Cacces Pathom Cours Cours Cours Cours Cours Cours Cours Cours Cours Cours Cours Cours Cou	Be6 приложение ASP.NET (NET Frameworld	Visual C#	Шаблони проектоя для создания принскимий XP/HIT Секи для Windre Law MT Transmick Вы полоте создавать страницая Вако: МИС. воб-АЯ и одностранитные приложения (SPA).				
	Имя: Proekt9							
	Расположение: C:\Users\serg	gel\source\repos	•	Обзор				
Стикок ошибок. Все решение - 😵 0 Ошибки 🛦 0 Предупре	Имя решения: Proekt9			Создать каталог для решения Создать новый репозиторий Git И ОК Отмена	• 4 × бок Р•			
Консоль дислетиера пакетов Сликок ошибок Вывод	o e e a		Ps 🐖	n 😨 🖬		Обозреватель решения	Team Explorer	8

Обратите внимание на «Настроить для HTTPS» - квадратик пуст.

Cozzata se6-novice ASP.NET.Core - Processo 7 X	
Asp. Marcore 2.1 Concertance and Concertance Asp. Marcore 2.1 Concertance and Concertance Asp. Marcore 2.1 Concertance and Con	
	slorer



Настройка стиля сайта

Выполните незначительную настройку меню, макета и домашней страницы сайта.

Откройте файл Views/Shared/_Layout.cshtml и внесите следующие изменения:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
   <title>@ViewData["Title"] - Proekt9</title>
   <environment include="Development">
       <link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.css" />
       <link rel="stylesheet" href="~/css/site.css" />
   </environment>
    <environment exclude="Development">
       k rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css"
             asp-fallback-href="~/lib/bootstrap/dist/css/bootstrap.min.css"
             asp-fallback-test-class="sr-only" asp-fallback-test-property="position"
asp-fallback-test-value="absolute" />
       <link rel="stylesheet" href="~/css/site.min.css" asp-append-version="true" />
   </environment>
</head>
<body>
   <nav class="navbar navbar-inverse navbar-fixed-top">
       <div class="container">
           <div class="navbar-header">
               <br/>button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".navbar-collapse">
                   <span class="sr-only">Toggle navigation</span>
                   <span class="icon-bar"></span>
                   <span class="icon-bar"></span>
                   <span class="icon-bar"></span>
               </button>
               <a asp-area="" asp-controller="Home" asp-action="Index" class="navbar-
brand">Proekt9</a>
           </div>
           <div class="navbar-collapse collapse">
               <a asp-area="" asp-controller="Home" asp-action="Index">Домашняя</a>
страница</a>
                   <a asp-area="" asp-controller="Students" asp-
action="Index">Студент</a>
                   <a asp-area="" asp-controller="Facultys" asp-
action="Index">Факультет</a>
                   <a asp-area="" asp-controller="Courses" asp-
action="Index">Kypc</a>
                   <a asp-area="" asp-controller="Trainers" asp-
action="Index">Преподаватель</a>
                   <a asp-area="" asp-controller="Grades" asp-
action="Index">Оценка</a>
               </div>
       </div>
   </nav>
```

```
<partial name="_CookieConsentPartial" />
<div class="container body-content">
    @RenderBody()
    <hr />
    <footer>
        &copy; 2020 - Proekt9
     </footer>
</div>
```

```
<environment include="Development">
        <script src="~/lib/jquery/dist/jquery.js"></script>
        <script src="~/lib/bootstrap/dist/js/bootstrap.js"></script>
        <script src="~/js/site.js" asp-append-version="true"></script>
    </environment>
    <environment exclude="Development">
        <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.3.1.min.js"</pre>
                asp-fallback-src="~/lib/jquery/dist/jquery.min.js'
                asp-fallback-test="window.jQuery"
                crossorigin="anonymous"
                integrity="sha384-
tsQFqpEReu7ZLhBV2VZ1Au7zcOV+rXbY1F2cqB8txI/8aZajjp4Bqd+V6D5IgvKT">
        </script>
        <script
src="https://stackpath.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"
                asp-fallback-src="~/lib/bootstrap/dist/js/bootstrap.min.js"
                asp-fallback-test="window.jQuery && window.jQuery.fn &&
window.jQuery.fn.modal"
                crossorigin="anonymous"
                integrity="sha384-
aJ210jlMXNL5UyI1/XNwTMqvzeRMZH2w8c5cRVpzpU8Y5bApTppSuUkhZXN0VxHd">
        </script>
        <script src="~/js/site.min.js" asp-append-version="true"></script>
    </environment>
   @RenderSection("Scripts", required: false)
</body>
</html>
```

Замените содержимое файла *Views/Home/Index.cshtml* следующим кодом, который заменяет текст о ASP.NET и MVC описанием этого приложения:

```
@{
    ViewBag.Title = "Home Page";
}
```

<h1 style="text-align:center; color:#796310">Курс ASP.NET Core MVC - создание базы данных</h1>

Создадим в папке Models пять сущностей (таблиц будущей базы данных) – тип файла класс Student , Faculty , Course , Trainer , Grades

И заменим сгенерированный код следующим:

```
using System.Collections.Generic;
using System.ComponentModel;
namespace Proekt9.Models
{
    public class Student
    {
        [DisplayName("Ключ")]
        public int StudentID { get; set; }
        [DisplayName("Фамилия")]
        public string FirstName { get; set; }
        [DisplayName("Имя")]
        public string LastName { get; set; }
                [DisplayName("Факультет")]
        11
                public int FacultyID { get; set; }
        11
        [DisplayName("Дата из")]
        public string DateFrom { get; set; }
        [DisplayName("Дата в")]
```

```
public string DateTo { get; set; }
        11
                 public virtual Faculty Faculty { get; set; }
        public virtual ICollection<Student> Students { get; set; } // включает одну
ссылку
    }
}
using System.Collections.Generic;
using System.ComponentModel;
namespace Proekt9.Models
{
    public class Faculty
    {
        [DisplayName("Ключ")]
        public int FacultyID { get; set; }
        [DisplayName("Название факультета")]
        public string FacultyTitle { get; set; }
        public virtual ICollection<Faculty> Facultys { get; set; } // Будет как
ссылачная таблица или классификатор
    }
}
using System.Collections.Generic;
using System.ComponentModel;
namespace Proekt9.Models
    public class Course
{
    {
                [DisplayName("Ключ")]
        public int CourseID { get; set; }
        [DisplayName("Название курса")]
        public string CourseTitle { get; set; }
        [DisplayName("Трайнер")]
        public int TrainerID { get; set; }
        [DisplayName("Факультет")]
        public int FacultyID { get; set; }
        [DisplayName("Дата в")]
        public string DateTo { get; set; }
        public virtual Trainer Trainer { get; set; }
        public virtual Faculty Faculty { get; set; }
        public virtual ICollection<Course> Courses { get; set; } // включает две ссылки
    }
}
using System.Collections.Generic;
using System.ComponentModel;
namespace Proekt9.Models
{
   public class Trainer
    {
        [DisplayName("Ключ")]
        public int TrainerID { get; set; }
```

```
[DisplayName("Фамилия")]
        public string FirstName { get; set; }
        [DisplayName("Имя")]
        public string LastName { get; set; }
[DisplayName("Дата из")]
        public string DataFrom { get; set; }
        [DisplayName("Дата в")]
        public string DateTo { get; set; }
        public virtual ICollection<Trainer> Trainers { get; set; } // Будет как
ссылачная таблица или классификатор
    }
}
using System.Collections.Generic;
using System.ComponentModel;
namespace Proekt9.Models
    public class Grade
{
    {
        [DisplayName("Ключ")]
        public int GradeID { get; set; }
        [DisplayName("Студент")]
        public int StudentID { get; set; }
        [DisplayName("Kypc")]
        public int CourseID { get; set; }
        [DisplayName("GradeValue")]
        public string GradeValue { get; set; }
        [DisplayName("Дата")]
        public string Date { get; set; }
        public virtual Course Course { get; set; }
        public virtual Student Student { get; set; }
        public virtual ICollection<Grade> Grades { get; set; } // включает две ссылки
   }
}
```

Создание контекста базы данных

Контекст базы данных — это основной класс, который координирует функциональные возможности Entity Framework для заданной модели данных. Этот класс создается путем наследования от класса Microsoft.EntityFrameworkCore.DbContext. В коде указываются сущности, которые включаются в модель данных. Также вы можете настроить реакцию платформы Entity Framework на некоторые события. В этом проекте соответствующий класс называется EFDbContext.

В папке проекта создайте папку Data.

В папке *Data* создайте новый файл класса с именем *EFDbContext.cs* и замените код шаблона следующим кодом:

```
using Proekt9.Models;
```

using Microsoft.EntityFrameworkCore;

```
namespace Proekt9.Data
{
   public class EFDbContext : DbContext
    {
        public EFDbContext(DbContextOptions<EFDbContext> options) : base(options)
        {
        }
        public DbSet<Course> Courses { get; set; }
        public DbSet<Faculty> Facultys { get; set; }
        public DbSet<Grade> Grades { get; set; }
        public DbSet<Student> Students { get; set; }
        public DbSet<Trainer> Trainers { get; set; }
        protected override void OnModelCreating(ModelBuilder modelBuilder)
        ł
            modelBuilder.Entity<Course>().ToTable("Course");
            modelBuilder.Entity<Faculty>().ToTable("Faculty");
            modelBuilder.Entity<Student>().ToTable("Student'
            modelBuilder.Entity<Grade>().ToTable("Grade");
            modelBuilder.Entity<Trainer>().ToTable("Trainer");
        }
   }
}
```

Регистрация EFDbContext

ASP.NET Core по умолчанию реализует технологию внедрения зависимостей. С помощью внедрения зависимостей службы (например, контекст базы данных EF) регистрируются во время запуска приложения. Затем компоненты, которые используют эти службы (например, контроллеры MVC), обращаются к ним через параметры конструктора. Код конструктора контроллера, который получает экземпляр контекста, будет приведен позднее в этом учебнике. Чтобы зарегистрировать EFDbContext как службу, откройте файл *Startup.cs* и добавьте выделенные строки в метод ConfigureServices.

```
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Proekt9.Data;
namespace Proekt9
{
    public class Startup
    {
        public Startup(IConfiguration configuration)
        {
            Configuration = configuration;
        }
}
```

```
public IConfiguration Configuration { get; }
        // This method gets called by the runtime. Use this method to add services to the
container.
        public void ConfigureServices(IServiceCollection services)
        {
            services.Configure<CookiePolicyOptions>(options =>
            ł
                // This lambda determines whether user consent for non-essential cookies
is needed for a given request.
                options.CheckConsentNeeded = context => true;
                options.MinimumSameSitePolicy = SameSiteMode.None;
            });
            services.AddDbContext<EFDbContext>(options =>
      options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection")));
            services.AddMvc().SetCompatibilityVersion(CompatibilityVersion.Version_2_1);
        }
        // This method gets called by the runtime. Use this method to configure the HTTP
request pipeline.
        public void Configure(IApplicationBuilder app, IHostingEnvironment env)
        {
            if (env.IsDevelopment())
            {
                app.UseDeveloperExceptionPage();
            }
            else
            {
                app.UseExceptionHandler("/Home/Error");
            }
            app.UseStaticFiles();
            app.UseCookiePolicy();
            app.UseMvc(routes =>
            {
                routes.MapRoute(
                    name: "default",
                    template: "{controller=Home}/{action=Index}/{id?}");
            });
       }
   }
}
```

Откройте файл *appsettings.json* и добавьте строку подключения, как показано в следующем примере.

```
{
    "ConnectionStrings": {
        "DefaultConnection":
    "Server=(localdb)\\mssqllocaldb;Database=Proekt91;Trusted_Connection=T
rue;MultipleActiveResultSets=true"
    },
    "Logging": {
        "IncludeScopes": false,
        "LogLevel": {
    }
}
```

```
"Default": "Warning"
}
}
```

SQL Server Express LocalDB

Строка подключения указывает на базу данных SQL Server LocalDB. LocalDB — это упрощенная версия ядра СУБД SQL Server Express, предназначенная для разработки приложений и не ориентированная на использование в производственной среде. LocalDB запускается по запросу в пользовательском режиме, поэтому настройки не слишком сложны.

Инициализация базы данных с тестовыми данными

Платформа Entity Framework создает пустую базу данных. В этом разделе вы напишете метод, который вызывается после создания базы данных и заполняет ее тестовыми данными.

Здесь будет использоваться метод EnsureCreated для автоматического создания базы данных.

В папке Data создайте новый файл класса с именем DbInitializer.cs и замените код шаблона следующим кодом, который обеспечивает создание базы данных и загрузку в нее тестовых данных в одну таблицу.

```
namespace Proekt9.Data
{
   public static class DbInitializer
        public static void Initialize(EFDbContext context)
            context.Database.EnsureCreated();
            // Look for any students.
            if (context.Students.Any())
            {
                return; // DB has been seeded
            }
           var students = new Student[]
            new Student{FirstName="Popov",LastName="Alexander",DateFrom = "11",DateTo = "22"},
           new Student{FirstName="Carson",LastName="Pavel",DateFrom = "111",DateTo = "222"},
            new Student{FirstName="Serov",LastName="Oleg",DateFrom = "1111",DateTo = "2222"}
            };
            foreach (Student s in students)
            {
                context.Students.Add(s);
            }
            context.SaveChanges();
       }
   }
```

В файле *Program.cs* измените метод Main, чтобы реализовать следующее поведение при запуске приложения:

- Получение экземпляра контекста базы данных из контейнера внедрения зависимостей.
- Вызов метода инициализации с передачей ему контекста.

}

• Высвобождение контекста после завершения работы метода заполнения.

```
public static void Main(string[] args)
{
     var host = CreateWebHostBuilder(args).Build();
    using (var scope = host.Services.CreateScope())
        var services = scope.ServiceProvider;
        try
        {
            var context = services.GetRequiredService<EFDbContext>();
            DbInitializer.Initialize(context);
        }
        catch (Exception ex)
        ł
            var logger = services.GetRequiredService<ILogger<Program>>();
            logger.LogError(ex, "An error occurred while seeding the database.");
        }
    }
    host.Run();
}
```

Теперь при первом запуске приложения будет создана и заполнена тестовыми данными необходимая для работы база данных. При любом изменении модели данных вы можете удалить базу, обновить метод заполнения и начать работу с новой базой данных аналогичным способом.

Создание контроллера и представлений

Далее вы будете использовать механизм шаблонов Visual Studio для добавления контроллера и представлений MVC, которые будут использовать платформу EF для запроса данных и их сохранения.

Автоматическое создание методов и представлений операций CRUD (создание, чтение, обновление и удаление) называется формированием шаблонов. Формирование шаблонов отличается от создания кода тем, что шаблонный код является отправной точкой и может изменяться в соответствии с потребностями, тогда как сформированный код обычно не изменяется. В тех случаях, когда требуется настроить созданный код в соответствии с внесенными изменениями, вы можете использовать разделяемые классы или повторно создать код.

 Щелкните правой кнопкой мыши папку Контроллеры в обозревателе решений и выберите Добавить > Создать шаблонный элемент.

ram.cs -# × Db	olnitializer.cs appsettings.json Startup.cs EFDbContext.cs Trainer. v ds Proekt9.Program	rs Student.cs Faculty.cs	Grade.cs Cou @ Main(string]] args)	rse.cs ErrorViewModeLcs	₹ 0602pesate	льрешений 🔫 🖡
17 E 18 19 20 21 E 22 23 24 E 25	<pre>public static void Main(string[] args) { var host = CreateWebHostBuilder(args).Build(); using (var scope = host.Services.CreateScope()) { var services = scope.ServiceProvider; try { } } }</pre>				Ofospesare Ofospesare Peruen Pro P P P P P P P P P P P P P P P P	ль решений — поих (Ctrl+ ие "Proekt9" (проектои: 1) sekt9 Connected Services Properties www.root Зависимости Состо Има
26 9 27 28 29 30	<pre>var context = services.GetRequiredServiceCETDEContext>(); DbInitializer.Initialize(context); } catch (Exception ex) {</pre>			 Просмотреть в браузере (Microsoft Edge) Диспетчер выполнения задач Настроить внешние инструменты Просмотреть с помощью 	Ctrl+Shift+W	ata * Dbinitializer.cs * EFDbContext.cs fodels
32 33 34 35 35 36 37 38 39 40 12 41 42 43 44 ↓ 45	<pre>iogger.logfron(ex, "An error occurred while seeding the da } host.Run(); } course i[oncomment public static TektHostBuilder CreateHobHostBuilder(string[] args) => Hebbost.CreateHobFaultBuilder(args) .UkeStartupcitartup>(); </pre>	Image: State State State Image: State I	Ctrl+Shift+A Shift+Alt+A	Zoosamb Crtps-n sevent as ropens of coperations Hoace npcgttasmene: Of coperations With Converts as ropenta X Buperatu X Komposetu X Apannin In Represensents Ontravers namer a mosequery Contravers namer a mosequery Contravers Contravers	, Ctrl+X Ctrl+C Del Alt+BBQ/Д	Home About celefini Contact celtrini Contact celtrini Contact celtrini Privacy, celtrini Shared Jeweingorts celtrini pasettings Low Jones rogram.cs artup.cs
6 асааны поиска 1 асааны поиска 1 асааны асе "Рг С:\Users\serge C:\Users\serge C:\Users\serge C:\Users\serge	X Yookt5", "Prodet5", Pesynkratw noseca 1, Ece oregunue googwernu Vloureelreposi/Prodet19/Prodet19/Node1s/Studer(c.6(6,11):namespace Prodet3.Hode1s Sloureelreposi/Prodet59/Prodet39/Node1s/Studer(c.6(6,11):namespace) Vloureelreposi/Prodet59/Node1s/Studer(c.6(6,11):namespace) Elsoureelreposi/Prodet59/Prodet39/Node1s/Course.c1(6,12):namespace Prodet3.Hode1s Elsoureelreposi/Prodet59/Prodet39/Node1s/Course.c1(6,12):namespace Prodet3.Mode1s				· + ×	

Image: Series Series Series Series Series Series Image: Series Image: Series Series Series Series Series Image: Series Image: Series Series Series Series Series Image: Series Image: Series Series Series Series Series Image: Series Series Series Series Series Series	entrating solation Colorado Colorado - 이 회원 18 19 19 19 19 10 10 10 10 10 ext.cs Trainer.cs Student.cs Faculty.cs Grade.cs C cogam · · · · · · · · · · · · · · · · · · ·	ourse.cs ErrorViewModeLcs	マ Обсиревстваь решений ・ キメ Үе ・ Обсиревстваь решений ・ キメ Үе ・ 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇
17 17 17 18 public static void Main(str) Addremve usbrous 19 19 19 19 Variose Variose 10 19 19 Variose Variose 21 10 using (var host = CreateWeblost Variosevice 21 10 using (var cope = host. 22 10 Try Observices = scope 23 10 Try Observices = scope 24 10 Contractions in the context = se 25 10 Context = se 26 10 Context = se 27 10 Context = se 28 10 Var logger = ser 30 10 Var logger - Logferord 31 10 Variosevice 33 10 Variosevice 34 10 Public static / NetHosthuille 40 10 Variosevice is set to the failBui 33 10 Variosevice is set to the failBui 34 10 Variosevice is set to the failBui 35 10 Variosevice is set to the failBui 36 10 Variosevice is set to the failBui 37 10 Variosevice is set to the failBui <t< td=""><td>Контроллер МVC с действилин чтении и записи Контроллер МVC - представлениями, использующий Entity Контроллер МVC – пустой Контроллер АРI – пустой Контроллер АРI – сдействиями, использующий Entity framework Контроллер АРI с действиями, использующий Entity framework</td><td>Контролер МVС с предсталениями, исловалурощий Entity Tranework по Макаросорт V1000 Контролер МVС с действиями и представителиями Rator для создания, сликов орщенствет из воэтеста данных Entity Fremework. Wde-melwaranoc MvCController/VithContex LStaffolder</td><td>Occupational and a second second</td></t<>	Контроллер МVC с действилин чтении и записи Контроллер МVC - представлениями, использующий Entity Контроллер МVC – пустой Контроллер АРI – пустой Контроллер АРI – сдействиями, использующий Entity framework Контроллер АРI с действиями, использующий Entity framework	Контролер МVС с предсталениями, исловалурощий Entity Tranework по Макаросорт V1000 Контролер МVС с действиями и представителиями Rator для создания, сликов орщенствет из воэтеста данных Entity Fremework. Wde-melwaranoc MvCController/VithContex LStaffolder	Occupational and a second
100 % → Beeoparatu noncca 1 ⓒ 교훈 웹 웹 결혼 X × = sawwartu ucc "Prodictos", "Prodicto", Peagynaratu no (c.Uker-Svergel Skourck/repos/ProdictO/ProdictO/Po	Шехоните здесь, чтобы выйти в Ингорнет и найти дополнительные расширения для формерования шаблонов.	Добавить Отиена	* * * * ×
C: Ulder: J serge1 Source / ropo 1/ horist // horist // horist // horist // serge1 Source / ropo 1/ horist // horist // horist // horist // serge1 Source / ropo 1/ horist // ho	ace Presty.Hodels ace Presty.Hodels © Prockty.Hodels ce Proekty.Models		• Обозреватель решения Теат Explorer

- В диалоговом окне **Добавление шаблона**:
- Выберите Контроллер MVC с представлениями, использующий Entity Framework.
- Нажмите кнопку **Добавить**. Откроется диалоговое окно **добавления** контроллера MVC с представлениями с использованием Entity Framework.

ram.cs -P X DbInitializer.cs appsettings.json Startup.cs EFDb0	Context.cs Trainer.cs Stud	ent.cs Faculty.cs Grade.cs	Course.cs ErrorViewModeLcs	₹ 00	озреватель решений 🚽 🖣 🗦
roekt9 * *** Proek	9.Program	* 🔍 Main(strir	ng[] args)		🖓 🖥 = "o = 🛱 🟠 😳
<pre>b</pre>		икениями, использующий Entity Framework week в задано в фейек Razer_viewstart) Доблаг	X V V T		Iorgentration, presente — monor (Chr
C:\Users\sergel\source\repos\ProektV\ProcktV\VocksV\Users\sergel\source\repos\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\Users\sergel\source\repos\ProektV\ProektV\VocksV\	espace Proekt9.Models pace Proekt9.Models space Proekt9.Models				

Заполняем форму

OBP/I Правка Вид Проект Сборка Отладка Команда Средства Тех Инстр O O O O O S	ументы R Анализ Окою С - ⊯ , § 15 05 15 12 potext.cs Trainer.cs	Cripaexa : 및 191 개 개 국 Student.cs F	aculty.cs Grade.cs	Course.cs E	ErrorViewModeLcs	Viktor Chas * VC
000000000000000000000000000000000000	плянск Тайлинск Program	Studentics F IngegicTauRenikamur, HCTIO Student (ProektS Moode UPDbContext (ProektS) Materia: Licational StudentsController StudentsController	Analyzes Grade at + @ Main(strin hasysoughi Entily Framework High Data) ee Razor_viewstarr) <u>AppSame</u>	Courtees F g] args)		Concrease the present Concrease the present Concrease the present Concrease services Concr
Спозет зает дел свои се отврия ут оексе (прексе утоексе утоекс	pace Proekts.Podeis					• Обозреватель решения Team Explorer
Этот элемент не поддерживает предварительный просмотр			Строка 26	Столбец 74 Знак	. 74 BCT	🛧 Добавить в систему управления версиями 🔺
🗖 🔎 Введите здесь текст для поиска 🛛 О 🗮 💽	🗖 🩋 🗖 📮	<i>ø 🔍</i> 👂	4 🔄 🔼 🖪	👱 🔒 🎙	s 🖌	へ 👄 🧟 (小)) ENG 12:34 🌄

При нажатии кнопки **Добавить** подсистема формирования шаблонов Visual Studio создает файл *StudentsController.cs* и набор представлений (файлы с расширением .*cshtml*), которые будут работать с контроллером.

