Asp Net 6

Using Visual Studio 2022 And Entity Framework

Search for templates (Alt+S)

Clear all

All languages   All platforms   Web

ASP.NET Core Web App
A project template for creating an ASP.NET Core application with example ASP.NET Razor Pages content.

C#   Linux   macOS   Windows   Cloud   Service   Web

Models/Coin.cs

```csharp
public class Coin
    {
        public int Id { get; set; }
        public string? Name { get; set; }
        public decimal Price { get; set; }

    }
```
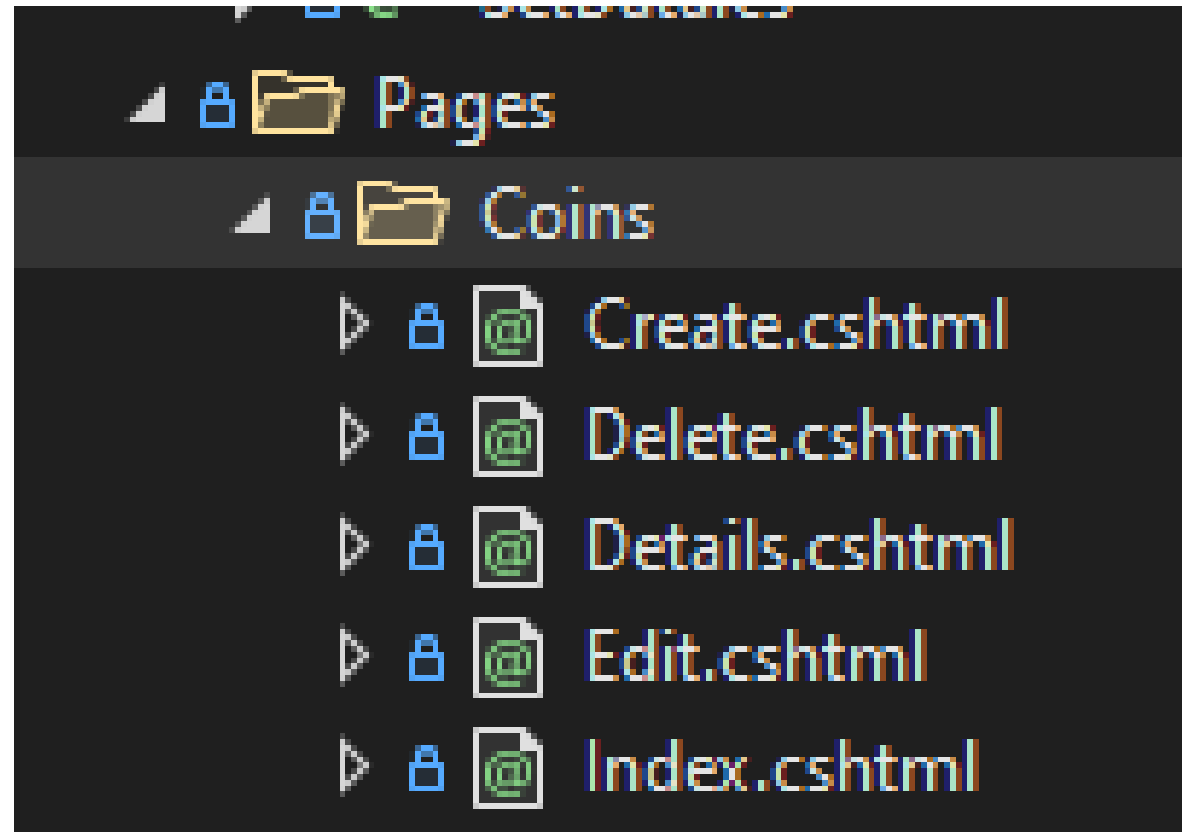
Установим доп модули

CRUT View

Add New Scaffolded Item

◢ Installed

◢ Common
  API
  ▷ MVC
  Razor Component
  Razor Pages
Identity
Layout

| @ Razor Page - Empty |
| @ Razor Page using Entity Framework |
| @ Razor Pages using Entity Framework (CRUD) |

Razor Pages using Entity Framework (CRUD)
by Microsoft
v1.0.0.0

Generates Razor Pages using Entity Framework for; Create, Delete, Details, Edit and List operations for the given model.

Id: ModelBasedCrudRazorPageScaffolder

Add Razor Pages using Entity Framework (CRUD)

Generates Razor Pages using Entity Framework for; Create, Delete, Details, Edit and List operations for the selected model.

Model class   Coin (CryptoDemo.Models)

Data context class   CryptoDemo.Data.CryptoDemoContext   +

Options

☐ Create as a partial view
☑ Reference script libraries
☑ Use a layout page

                                                              ...

(Leave empty if it is set in a Razor _viewstart file)

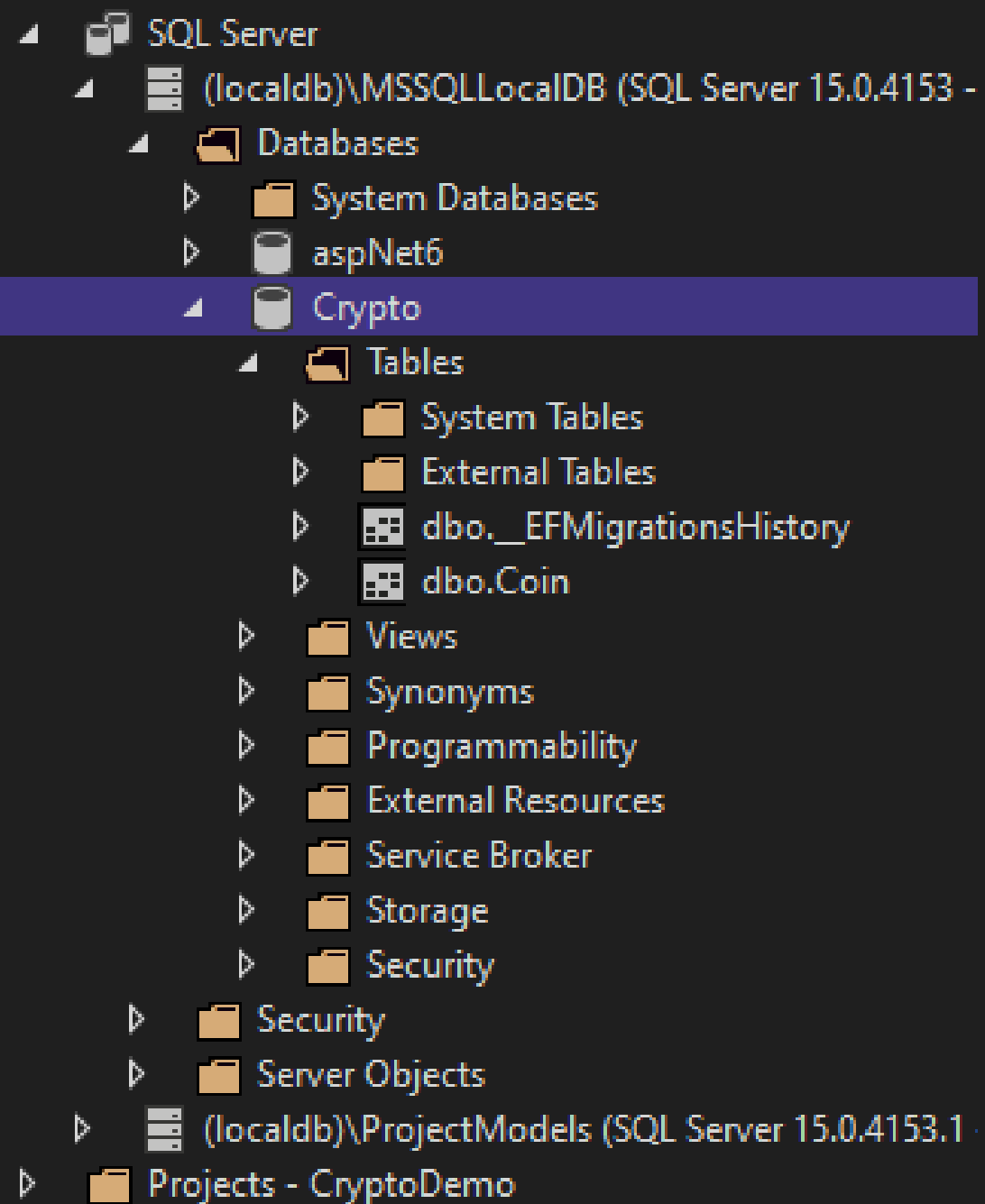   Add   Cancel

```
Type 'get-help NuGet' to see all available NuGet commands.

PM> add-migration InitData
```

```
Package Manager Console Host Version 6.1.0.106

Type 'get-help NuGet' to see all available NuGet commands.

PM> update-database
```

- ▲ 🗄 SQL Server
  - ▲ 🗄 (localdb)\MSSQLLocalDB (SQL Server 15.0.4153 -
    - ▲ 📁 Databases
      - ▷ 📁 System Databases
      - ▷ 🗃 aspNet6
      - ▲ 🗃 Crypto
        - ▲ 📁 Tables
          - ▷ 📁 System Tables
          - ▷ 📁 External Tables
          - ▷ 🎹 dbo._EFMigrationsHistory
          - ▷ 🎹 dbo.Coin
        - ▷ 📁 Views
        - ▷ 📁 Synonyms
        - ▷ 📁 Programmability
        - ▷ 📁 External Resources
        - ▷ 📁 Service Broker
        - ▷ 📁 Storage
        - ▷ 📁 Security
    - ▷ 📁 Security
    - ▷ 📁 Server Objects
  - ▷ 🗄 (localdb)\ProjectModels (SQL Server 15.0.4153.1 -
- ▷ 📁 Projects - CryptoDemo

Name	Data Type	Allow Nulls	Default
📌 Id	int	☐	
Name	nvarchar(MAX)	☑	
Price	decimal(18,2)	☐	
		☐	

▲ **Keys** (1)
  PK_Coin (Primary Key, Clustered: Id)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

🖳 Design | ↑↓ | 🖳 T-SQL

```sql
1   CREATE TABLE [dbo].[Coin] (
2       [Id]    INT             IDENTITY (1, 1) NOT NULL,
3       [Name]  NVARCHAR (MAX)  NULL,
4       [Price] DECIMAL (18, 2) NOT NULL,
5       CONSTRAINT [PK_Coin] PRIMARY KEY CLUSTERED ([Id] ASC)
6   );
7
```

144 % ▾ | ✅ No issues found | ◀ | Ln: 8 Ch: 1 TABS MIXED

Models/SetData.cs

```csharp
using CryptoDemo.Data;
using Microsoft.EntityFrameworkCore;

namespace CryptoDemo.Models
{
    public static class SetData
    {
        public static void Initialize(IServiceProvider serviceProvider)
        {
            using (var context = new CryptoDemoContext(
                serviceProvider.GetRequiredService<
                    DbContextOptions<CryptoDemoContext>>()))
            {
                if (context == null || context.Coin == null)
                {
                    throw new ArgumentNullException("Null CryptoDemoContext");
                }


                if (context.Coin.Any())
                {
                    return;
                }
```

```csharp
            context.Coin.AddRange(
                new Coin
                {
                    Name = "Bitcoin",
                    Price = 42920
                },

                new Coin
                {
                    Name = "Litecoin",
                    Price = 570
                },

                new Coin
                {
                    Name = "Dogecoin",
                    Price = 23
                },

                new Coin
                {
                    Name = "BNB",
                    Price = 450
                }
            );
            context.SaveChanges();
        }
    }
}
```

Program.cs update

```
var app = builder.Build();

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

    SetData.Initialize(services);
}
```

dbo.Coin [Data] ⚲ ✕ CryptoDemoContext.cs

Max Rows: 1000 ▾

	Id	Name	Price
▷	1	Bitcoin	42920.00
	2	Litecoin	570.00
	3	Dogecoin	230.00
	4	BNB	450.00
	5	Fanis	1000000.00
✱	NULL	NULL	NULL

_Layout.cshtml update

```html
<div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">
    <ul class="navbar-nav flex-grow-1">
        <li class="nav-item">
            <a class="nav-link text-dark" asp-area="" asp-page="/Index">Home</a>
        </li>
         <li class="nav-item">
            <a class="nav-link text-dark" asp-area="" asp-page="/Coins/Index">Coins</a>
        </li>
        <li class="nav-item">
            <a class="nav-link text-dark" asp-area="" asp-page="/Privacy">Privacy</a>
        </li>
    </ul>
</div>
```