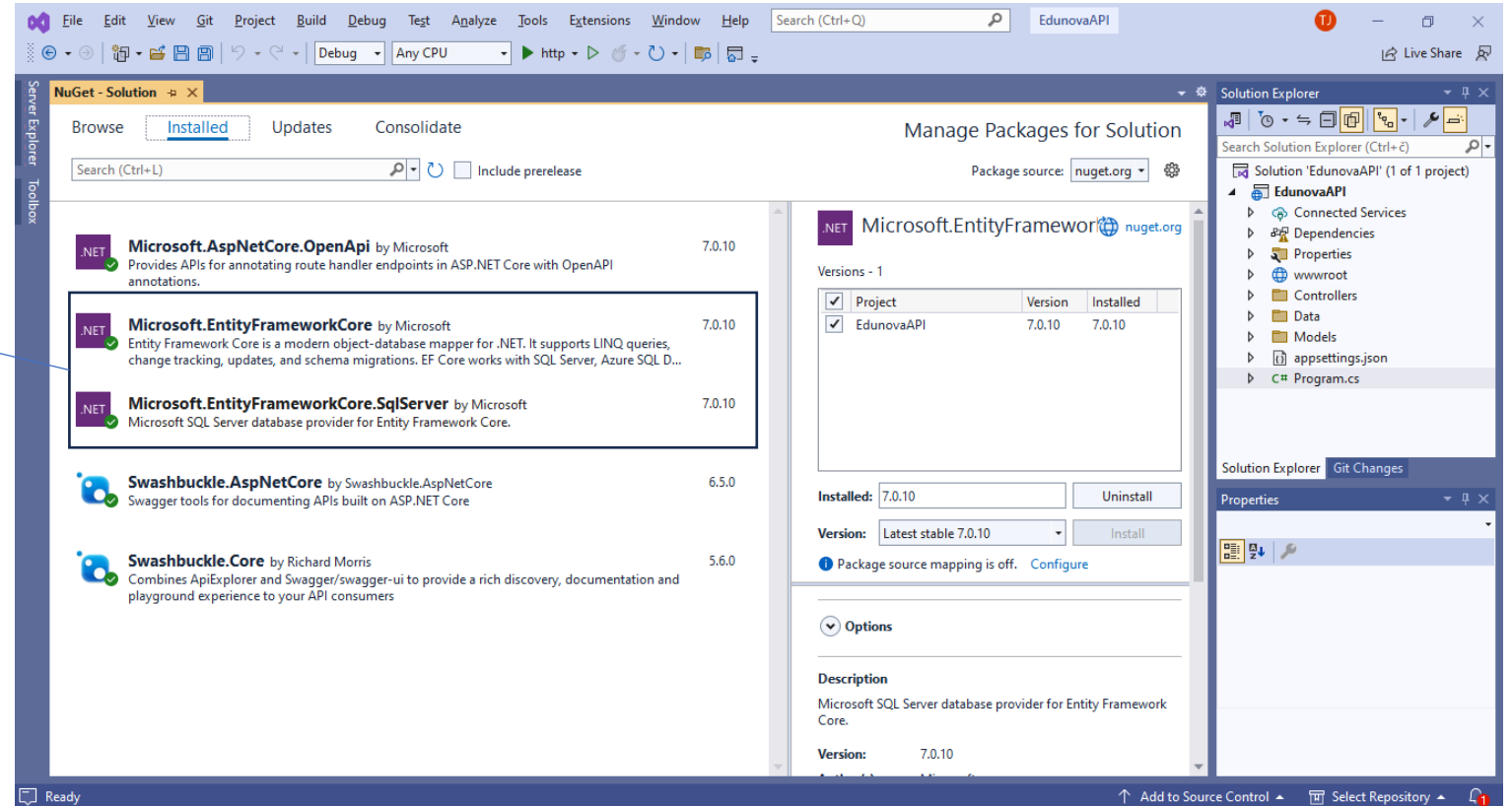


Web API bootstrap + rad s bazom

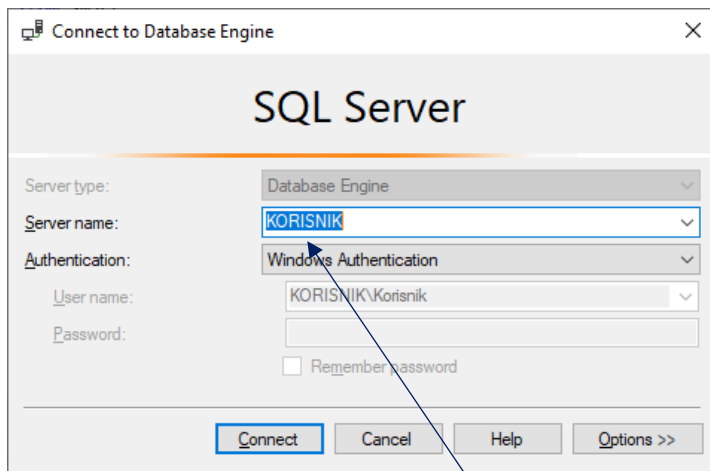
Potrebno je dodati EntityFramework
(SQL server) zavisnosti



Izvori:

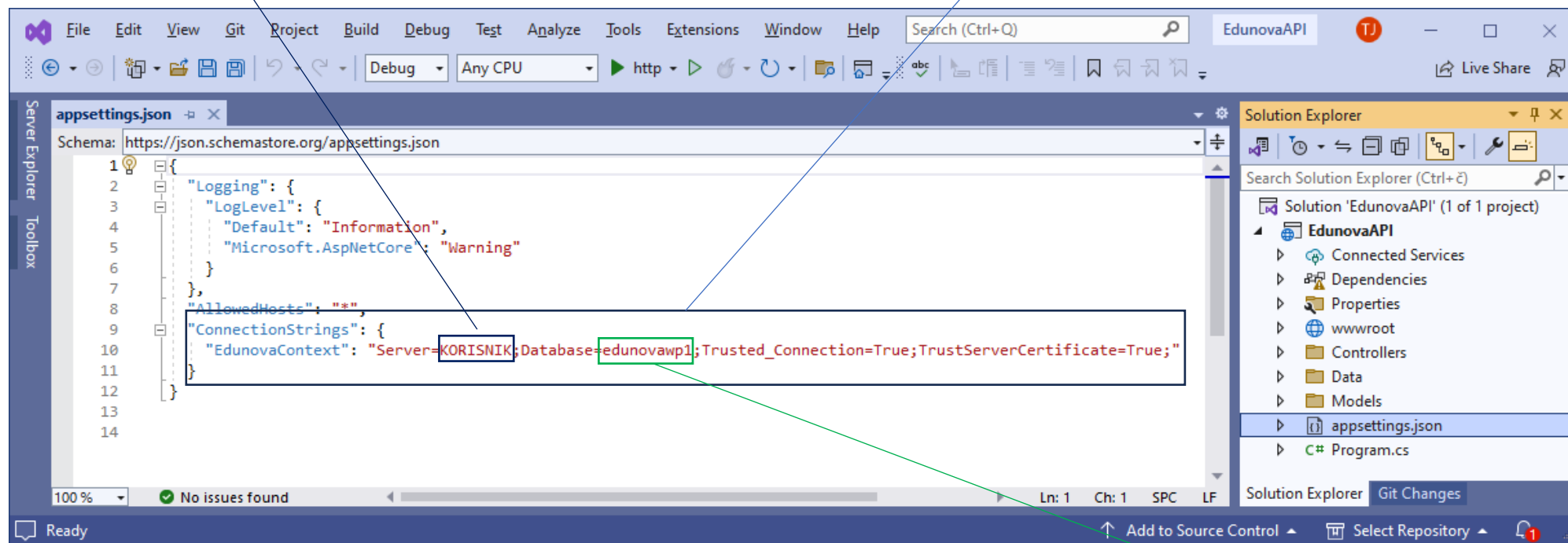
<https://learn.microsoft.com/en-us/ef/>

<https://www.entityframeworktutorial.net/what-is-entityframework.aspx>



Web API bootstrap + rad s bazom

U datoteci appsettings.json je potrebno definirati Connection string gdje se nalazi baza



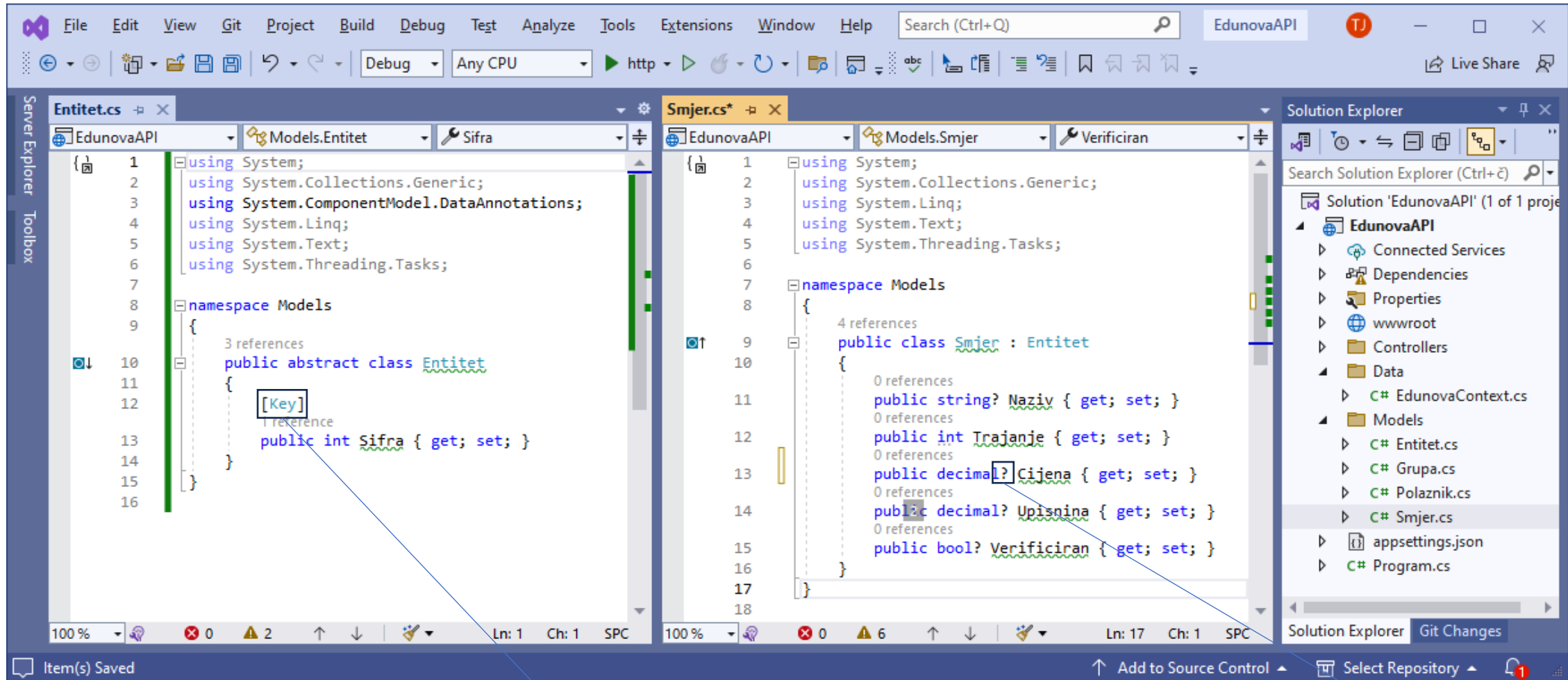
Izvor:

<https://www.connectionstrings.com/sql-server/>

Ime baze

Web API bootstrap + rad s bazom

Potrebno je prilagoditi postojeće model klase



Izvori: Moramo postaviti atribut Key da EF zna što je primarni ključ

Moramo postaviti mogućnost null vrijednosti (?)

<https://learn.microsoft.com/en-us/ef/core/modeling/keys?tabs=data-annotations>

<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/builtin-types/nullable-reference-types>

Web API bootstrap + rad s bazom

Potrebno je kreirati datoteku EdunovaContext koja nasljeđuje DbContext

The screenshot displays the Visual Studio IDE with the EdunovaContext.cs file open. The code defines the EdunovaContext class, which inherits from DbContext. It includes a DbSet<Smjer> property named smjer. The SQL Server Enterprise Edition interface shows the edunovawp1 database with tables dbo.clan, dbo.grupa, dbo.polaznik, and dbo.smjer. The query results show a table with columns sifra, naziv, trajanje, cijena, upisnina, and verificiran.

```
using Microsoft.EntityFrameworkCore;
using Models;

namespace EdunovaAPI.Data
{
    public class EdunovaContext : DbContext
    {
        public EdunovaContext(DbContextOptions<EdunovaContext> options)
            : base(options)
        {
        }

        public DbSet<Smjer> smjer { get; set; }
    }
}
```

Object Explorer shows the database structure:

- KORISNIK (SQL Server 16.0.1050.5 - KORISNIK\Korisnik)
- Databases
 - System Databases
 - Database Snapshots
 - edunovawp1
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.clan
 - dbo.grupa
 - dbo.polaznik
 - dbo.smjer
 - Dropped Ledger Tables

SQLQuery1.sql - KORISNIK.edunovawp1 (KORISNIK.Korisnik (58))* - Microsoft SQL Server

```
select * from smjer;
```

| | sifra | naziv | trajanje | cijena | upisnina | verificiran |
|---|-------|----------------------|----------|---------|----------|-------------|
| 1 | 1 | Web programiranje ŠČ | 250 | 1473.82 | 73.00 | 1 |
| 2 | 2 | Java programiranje | 130 | NULL | NULL | NULL |
| 3 | 3 | Serviser | 150 | NULL | NULL | 0 |

Results | Messages

RISNIK (16.0 RTM) | KORISNIK\Korisnik (58) | edunovawp1 | 00:00:00 | 3 rows

svojstvo DbSet<Smjer> se mora zvati kao tablica u bazi

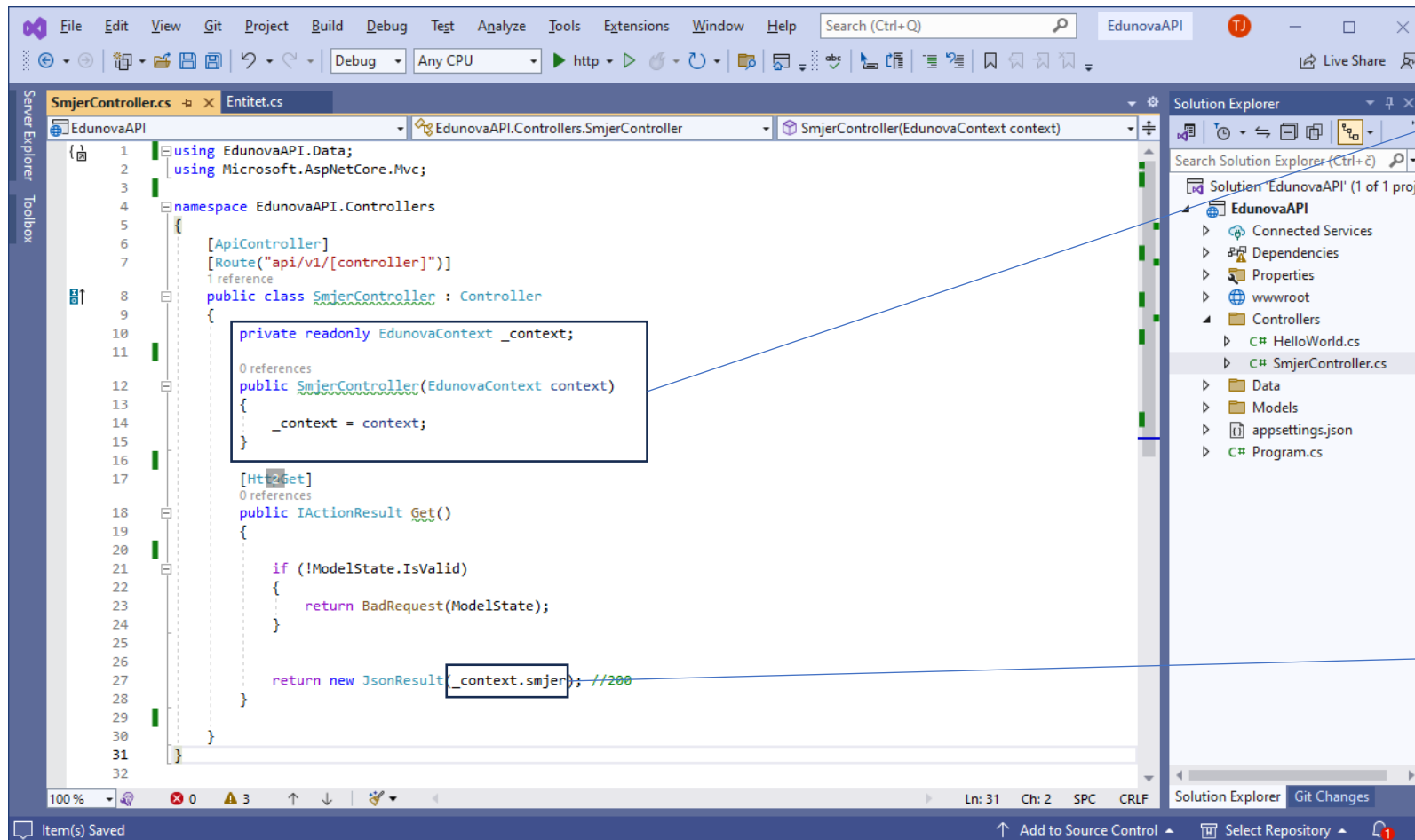
Izvori:

<https://learn.microsoft.com/en-us/ef/>

<https://www.entityframeworktutorial.net/what-is-entityframework.aspx>

Web API bootstrap + rad s bazom

Korištenje DbContext u našem kontroleru (GET)



Izvori:

<https://learn.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection?view=aspnetcore-7.0>

<https://www.tutorialsteacher.com/core/dependency-injection-in-aspnet-core>

Testiranje GET rute

Smjer

Web API bootstrap + rad s bazom

GET /api/v1/Smjer

Parameters

Cancel

No parameters

ExecuteClear

Responses

Response content type: application/json

Snippets

cURL (bash)cURL (PowerShell)cURL (CMD)

```
curl -X 'GET' \
'http://localhost:7191/api/v1/Smjer' \
-H 'accept: application/json'
```

Request URL

http://localhost:7191/api/v1/Smjer

Server response

| Code | Details |
|------|---|
| 200 | <div>Response body<pre>[{ "naziv": "Web programiranje ŠČ", "trajanje": 250, "cijena": 1473.82, "upisnina": 73, "verificiran": true, "sifra": 1 }, { "naziv": "Java programiranje", "trajanje": 130, "cijena": null, "upisnina": null, "verificiran": null, "sifra": 2 }, { "naziv": "Serviser", "trajanje": 150, "cijena": null, "upisnina": null, "verificiran": false, "sifra": 3 }]</pre></div> <div>Response headers<pre>content-type: application/json; charset=utf-8 date: Sun, 20 Aug 2023 13:24:43 GMT server: Kestrel transfer-encoding: chunked</pre></div> |

Responses

Web API bootstrap + rad s bazom

Korištenje DbContext u našem kontroleru (POST)

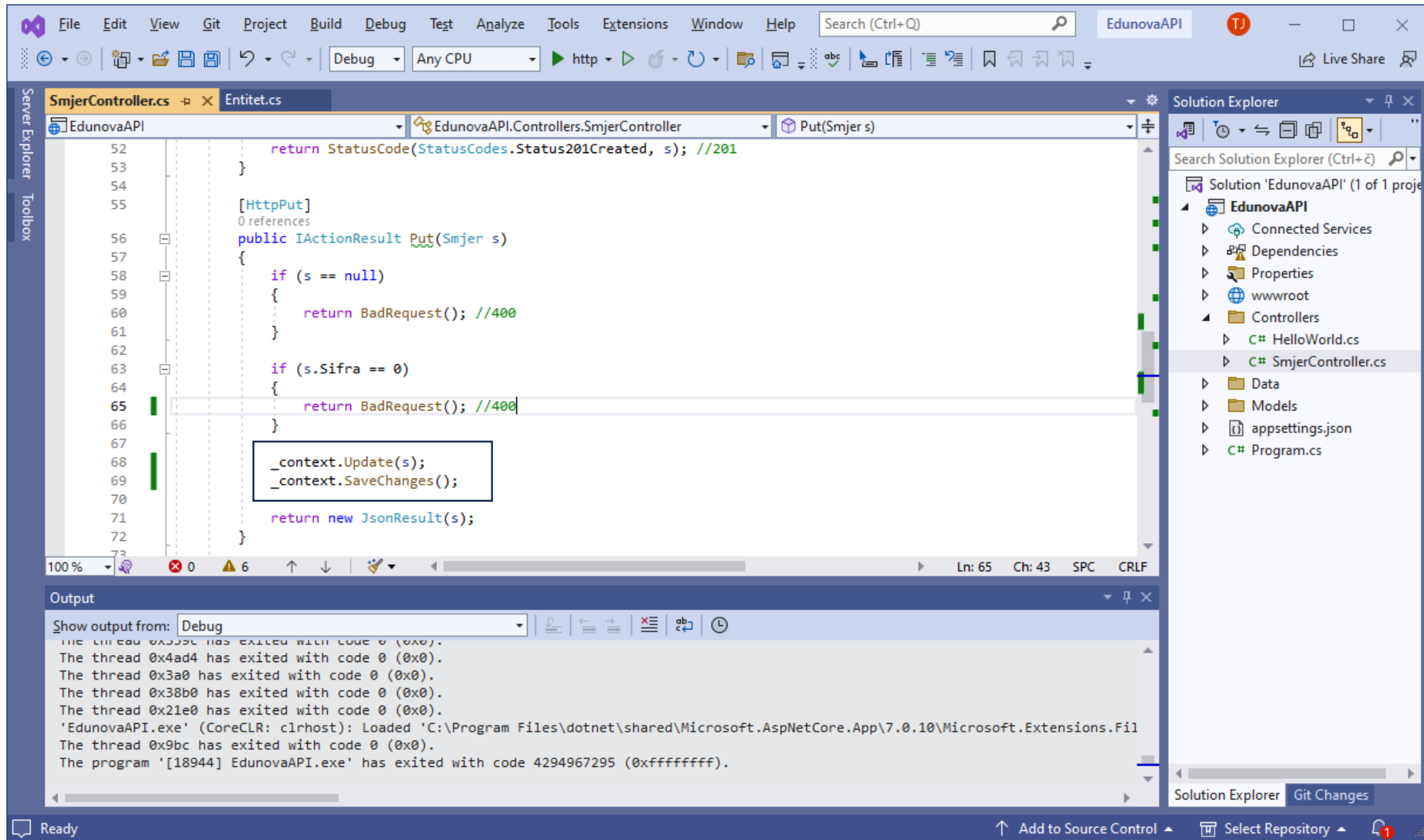
The screenshot displays the Visual Studio IDE with the following components:

- Code Editor:** Shows the `SmjerController.cs` file. The `Post` method is highlighted, showing a call to `_context.Add(s);` and `_context.SaveChanges();`. The method signature is `public IActionResult Post(Smjer s)`.
- Diagnostic Tools:** Located on the right, it shows a diagnostics session of 1:58 minutes. The `Events` and `Process Memory (MB)` tabs are visible. The `Process Memory (MB)` tab shows a memory usage of 62 MB.
- Autos:** Located at the bottom left, it shows a search for `Ctrl+E` with a search depth of 3.
- Call Stack:** Located at the bottom right, it shows a search for `Ctrl+E` and a view of all threads.
- Bottom Bar:** Includes tabs for `Autos`, `Locals`, `Watch 1`, `Call Stack`, `Breakpoints`, `Exception Settings`, `Command Window`, `Immediate Window`, and `Output`.

```
37  
38     return new JsonResult(_context.smjer); //200  
39 }  
40  
41 [HttpPost]  
42 0 references  
43 public IActionResult Post(Smjer s)  
44 {  
45     if(s == null)  
46     {  
47         return NoContent(); // 204  
48     }  
49  
50     _context.Add(s);  
51     _context.SaveChanges();  
52  
53     return StatusCode(StatusCode.Status201Created, s); //201  
54 }  
55  
56 [HttpPut]  
57 0 references
```

Web API bootstrap + rad s bazom

Korištenje DbContext u našem kontroleru (PUT)



Web API bootstrap + rad s bazom

Korištenje DbContext u našem kontroleru (DELETE)

