Connecting the ASP.NET Core Web Application to SQL Server

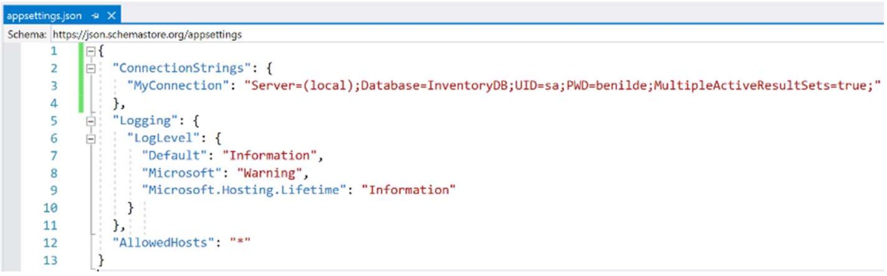
1. From the **Solution Explorer**, open **appsettings.json**.
2. Include the connection string property and its value.

   Using SQL Server Authentication

   a. Server = (local) or computer name
   b. Database = InventoryDB or equivalent
   c. UID = sa
   d. PWD / Password = benilde
   e. MultipleActiveResults = true

   Using Windows Authentication

   a. Server = (local)
   b. Database = Inventory
   c. Integrated Security = true
   d. MultipleActiveResultSets = true

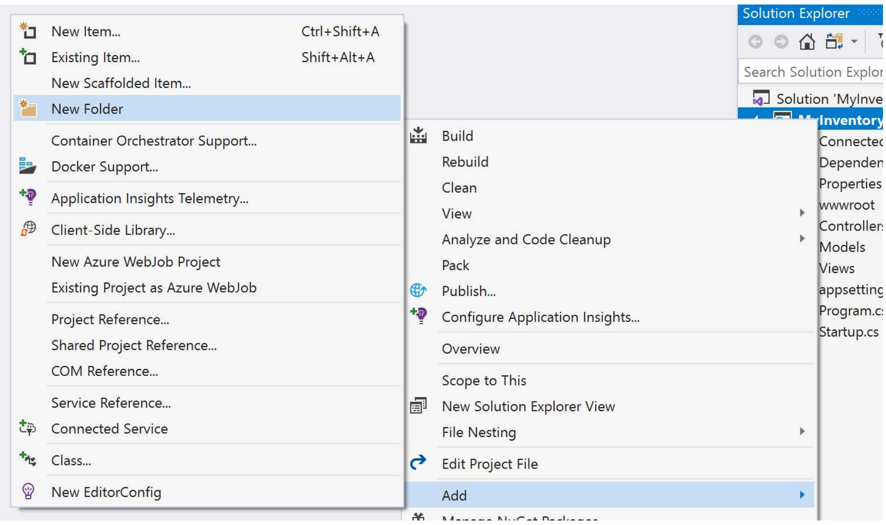


appsettings.json

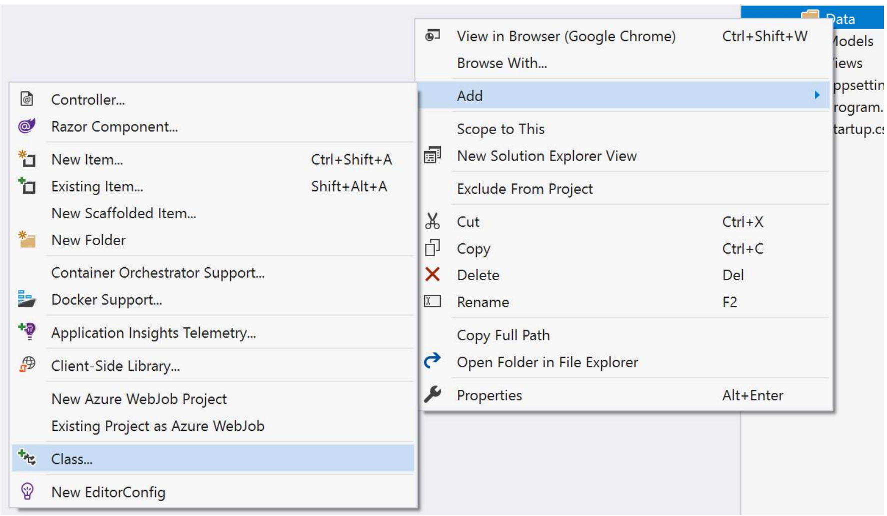
Schema: https://json.schemastore.org/appsettings

```
{
  "ConnectionStrings": {
    "MyConnection": "Server=(local);Database=InventoryDB;UID=sa;PWD=benilde;MultipleActiveResultSets=true;"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedHosts": "*"
}
```

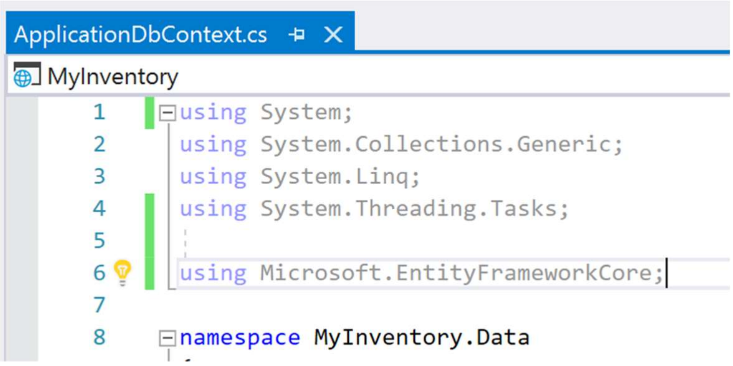
3. Save the file.
4. From the **Solution Explorer**, right-click the project, then select **Add > New Folder**.



5. Rename the newly added folder to **Data**.
6. Select **Data** folder then right-click **Add > Class...**

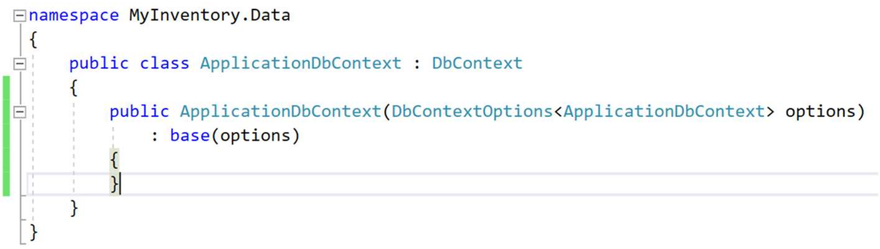


7. From the **Add New Item** window, name it **ApplicationDbContext.cs**.
8. From **ApplicationDbContext.cs**, add the namespace **Microsoft.EntityFrameworkCore**.



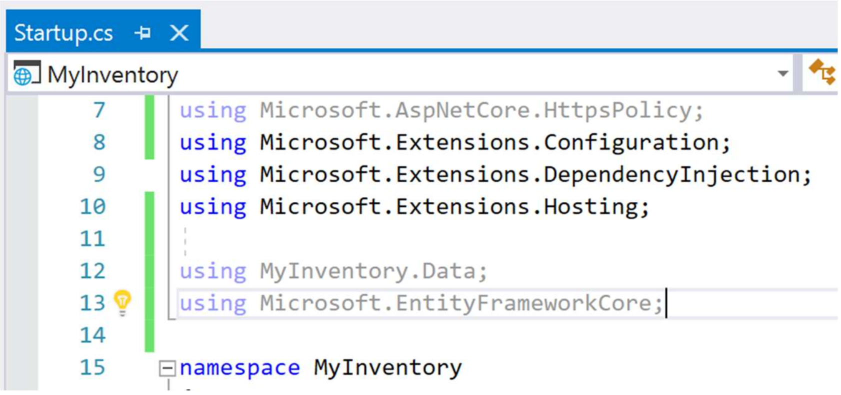
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

using Microsoft.EntityFrameworkCore;

namespace MyInventory.Data
```

9. Update **ApplicationDbContext.cs** to derive **ApplicationDBContext** class from Entity Framework Core’s **DbContext** class..



```
namespace MyInventory.Data
{
    public class ApplicationDbContext : DbContext
    {
        public ApplicationDbContext(DbContextOptions<ApplicationDbContext> options)
            : base(options)
        {
        }
    }
}
```

10. Save the file.
11. From the **Solution Explorer**, open **Startup.cs**.
12. Add the namespaces **MyInventory.Data** and **Microsoft.EntityFrameworkCore**.



```
using Microsoft.AspNetCore.HttpsPolicy;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;

using MyInventory.Data;
using Microsoft.EntityFrameworkCore;

namespace MyInventory
```

13. Update the **Startup.ConfigureServices** method to use the Entity Framework service.

```
// This method gets called by the runtime. Use this method to add services to the container.
public void ConfigureServices(IServiceCollection services)
{
    services.AddDbContext<ApplicationDbContext>(option =>
        option.UseSqlServer(Configuration.GetConnectionString("MyConnection")));
    services.AddControllersWithViews();
}
```

14. Save the file.