

[Open in app](#)

Medium

 Search

★ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Connecting MySql to .NET core app

MySQL connection in .NET Core 8 MVC



Mohammed Taherali · [Follow](#)

3 min read · Feb 16, 2024



Listen



Share



More

Setting up Database in .NET maybe a daunting task, however, this tutorial will walk you through each step to setup a seamless connection.

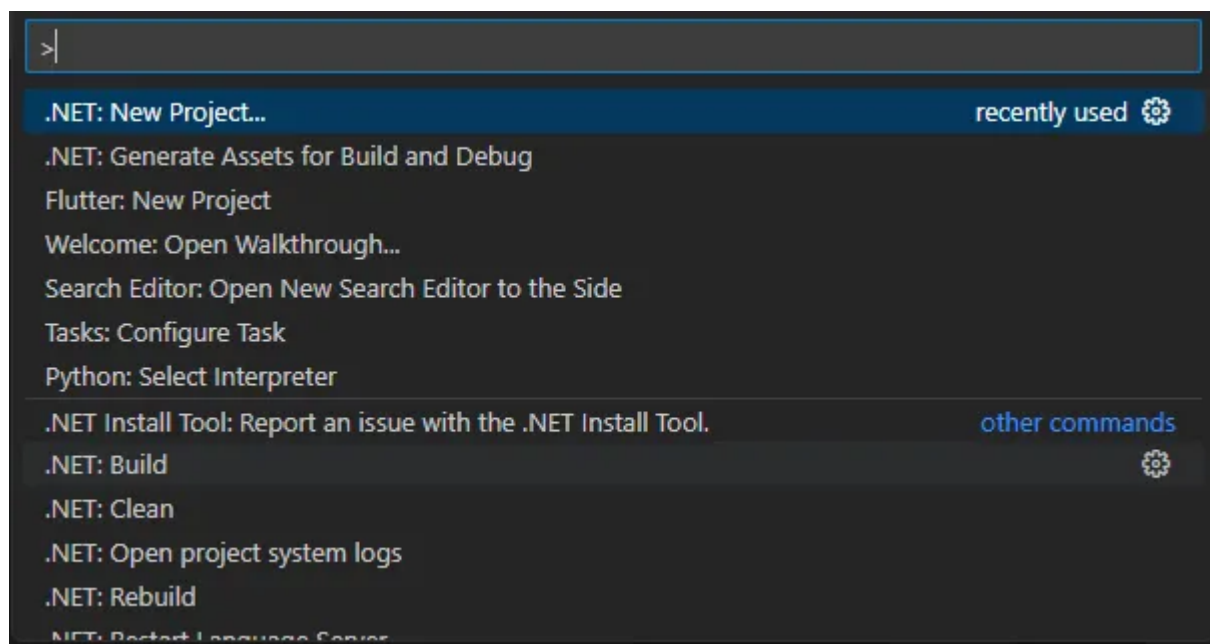
Prerequisites:

1. MySql server and Mysql workbench [Follow along [this tutorial](#) to set it up

2. .NET core 8 (see [here](#))

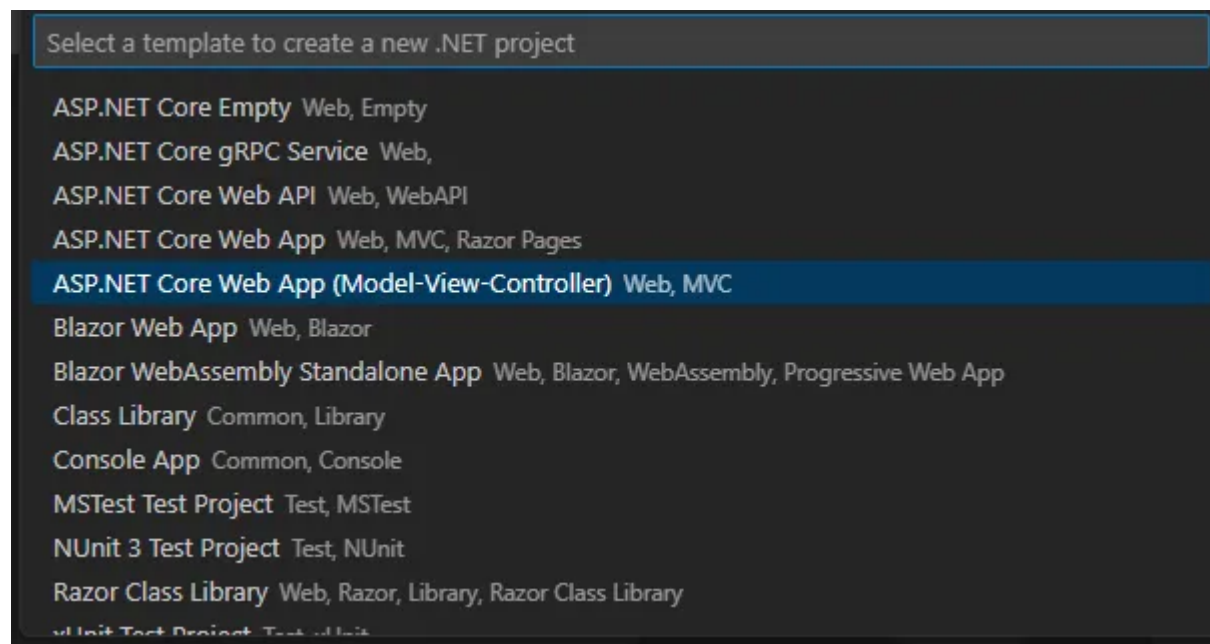
Create a .NET core MVC app

Open command palette using 'Ctrl + Shift + p' and create a ".NET: New Project..."



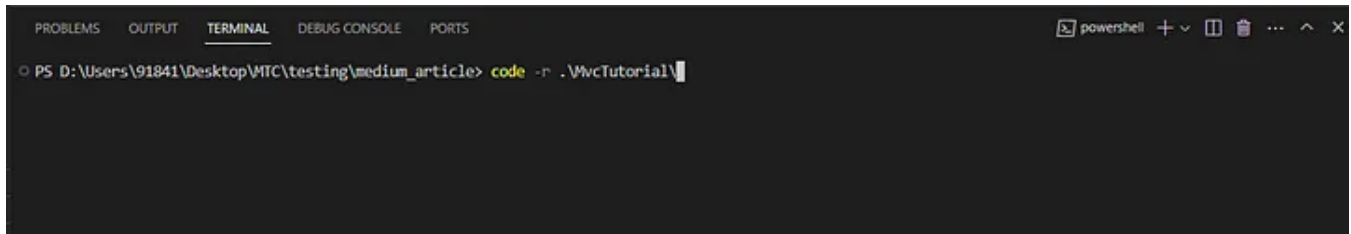
Create .NET new project

Select ASP.NET Core MVC app



Select MVC app

open the folder in the current instance of Vscode



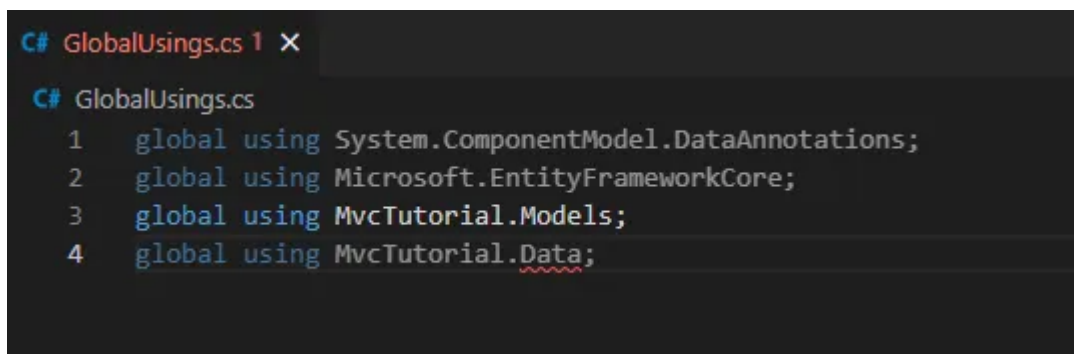
replace the folder in current instance of vscode

Install the necessary Nuget packages using the terminal

```
dotnet tool install --global dotnet-aspnet-codegenerator
dotnet tool install --global dotnet-ef
dotnet add package Microsoft.EntityFrameworkCore
dotnet add package Microsoft.EntityFrameworkCore.Tools
dotnet add package Pomelo.EntityFrameworkCore.MySql --version 8.0.0
```

Make sure the pomelo SQL Nuget package's version matches the entity framework core and tools (in this case version 8)

create a GlobalUsings.cs file in the root directory. This allows us to state all the import packages in one file so that we don't have to import them in every single file.



GlobalUsings.cs file

For now, don't worry about the error. We will shortly create the Data folder.

Creating a model

Create a file named *Users.cs* inside the Models folder

```
1 namespace MvcTutorial.Models
2 {
3     public class Users
4     {
5         [Key]
6         public int UserId { get; set; }
7
8         [Required]
9         public string? Name { get; set; }
10
11         public int Age { get; set; }
12     }
13 }
```

Users.cs hosted with ❤ by GitHub

[view raw](#)

Users.cs file

Create a new folder in the root directory named *'Data'* and create a new class file named *"ApplicationDbContext.cs"*

```
1 namespace MvcTutorial.Data
2 {
3     public class ApplicationDbContext : DbContext
4     {
5         public ApplicationDbContext(DbContextOptions<ApplicationDbContext> options) : base(options)
6         {
7         }
8
9         public DbSet<Users> Users { get; set; }
10     }
11 }
```

ApplicationDbContext.cs hosted with ❤ by GitHub

[view raw](#)

ApplicationDbContext.cs

Now let's add the connection string for the MySQL server.

Open *"appsettings.json"* file in the root directory and add the following part

```
appsettings.json X
appsettings.json > ...
1 {
2   "Logging": {
3     "LogLevel": {
4       "Default": "Information",
5       "Microsoft.AspNetCore": "Warning"
6     }
7   },
8   "AllowedHosts": "*",
9   "ConnectionStrings": {
10    "DefaultConnection": "Server=localhost;Database=mvctutorial;User=root;Password=password;"
11  }
12 }
13
```

appsettings.json file

you change the database name as you like.

Now, for the final code change, open your “*Program.cs*” file in root directory and add the following lines

```
Program.cs X
Program.cs
1 var builder = WebApplication.CreateBuilder(args);
2
3 // Add services to the container.
4 builder.Services.AddControllersWithViews();
5
6 var connectionString = builder.Configuration.GetConnectionString("DefaultConnection");
7 builder.Services.AddDbContext<ApplicationDbContext>(options =>
8 {
9     options.UseMySQL(connectionString, ServerVersion.AutoDetect(connectionString));
10 });
11
12 var app = builder.Build();
13 // Configure the HTTP request pipeline.
14 if (!app.Environment.IsDevelopment())
15 {
16     app.UseExceptionHandler("/Home/Error");
17     // The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/asp
18     app.UseHsts();
19 }
20
21 app.UseHttpsRedirection();
22 app.UseStaticFiles();
23 app.UseRouting();
24 app.UseAuthorization();
25
26 app.MapControllerRoute(
27     name: "default",
28     pattern: "{controller=Home}/{action=Index}/{id?}");
29
30 app.Run();
31
```

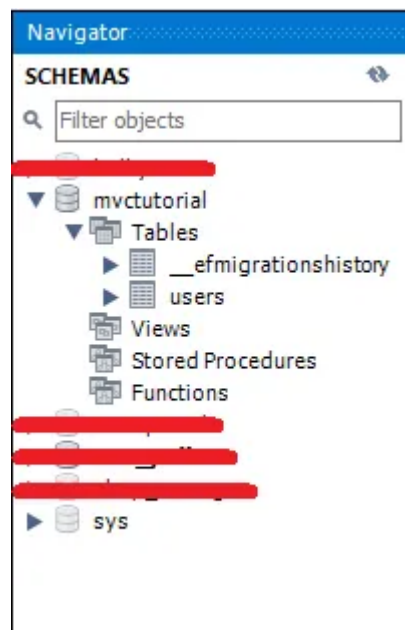
Finally, Add Migrations and create the database

In the console run the following command to create the migrations file and also create the database.

```
dotnet ef migrations add init
dotnet ef database update
```

You will see that a folder named “*Migrations*” will be created, which is for initially setting up the database and tables.

In your MySql workbench, you will find the new database named ‘*mvctutorial*’ created.



mvctutorial database created

That's it for today's tutorial!

Follow [Mohammed Taherali](#) for more such tutorials!

Connect with me on [LinkedIn](#)

Like, Share and comment!

MySQL

Net Core

Database Connectivity

Microsoft

Web Development

[Follow](#)

Written by Mohammed Taherali

5 Followers · 6 Following

Responses (1)



Fabianojd Pos

What are your thoughts?



Kutay Sutcu

Feb 7




Thank you. It was very clear.



[Reply](#)

More from Mohammed Taherali




 Mohammed Taherali

Authenticate user using Nodejs and Expressjs

Authenticating users is the first step of any software system. So it is crucial to have a robust authentication system available to prevent...

Sep 3, 2023  3  1




 Mohammed Taherali

Linear regression: Mini batch gradient descent from scratch

In this tutorial we will be creating our own mini batch gradient descent algorithm from scratch using python.

Sep 4, 2022 🖱 1

 Mohammed Taherali

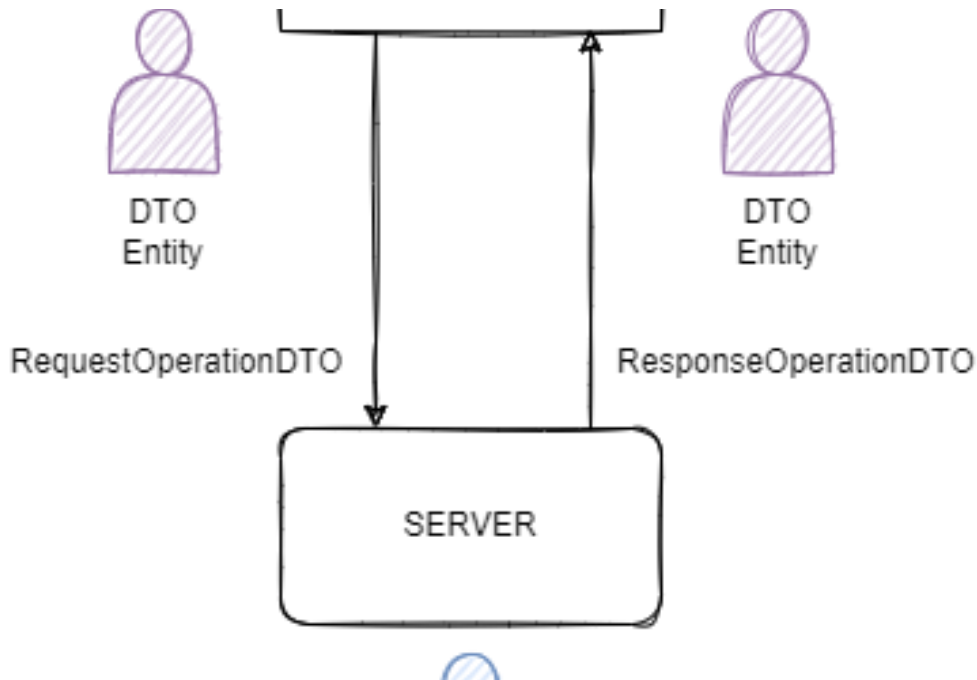
Linear Regression: From scratch in python


Previously we saw how to implement gradient descent algorithm ([click here](#)). Now we will see how to implement complete linear regression...

Sep 7, 2022

[See all from Mohammed Taherali](#)

Recommended from Medium



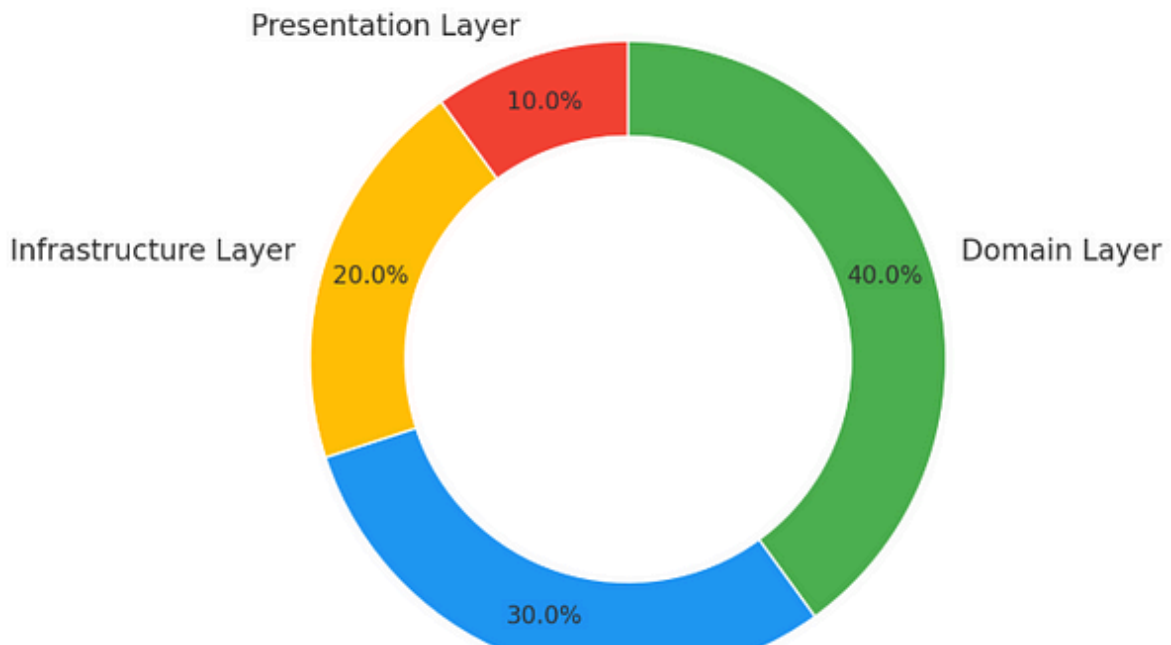
 Krish Panchal


The Role of DTOs in RESTful .NET APIs: A Complete Guide

While building RESTful APIs in .NET, it's essential to expose only the necessary data, prevent overposting or underposting, and minimize...

✦ 3d ago 🖱 14 💬 1

🔖+ ...

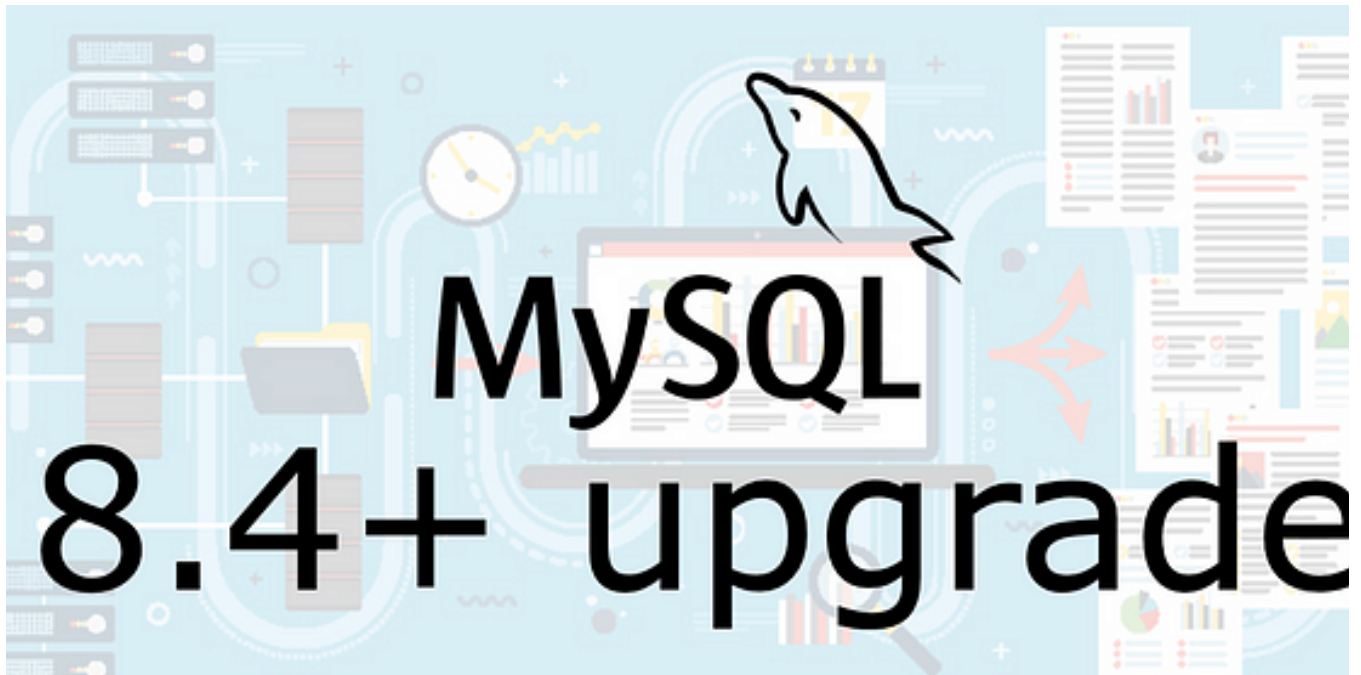


 AJ Software Engineering

Studying Clean Architecture in .NET 8 as a Laravel Engineer

As a Laravel engineer, switching to .NET and learning Clean Architecture is pretty interesting and has given me a new perspective on how...

Jan 29 🖱️ 2



Alexander Sharov

Fix mysql_native_password not loaded errors in MySQL 8.4+

After upgrading to MySQL 8.4 or Bitnami's MySQL Helm chart version 11.0.0 and above, many users encountered the following error:

★ Oct 25, 2024 🖱️ 4 💬 1

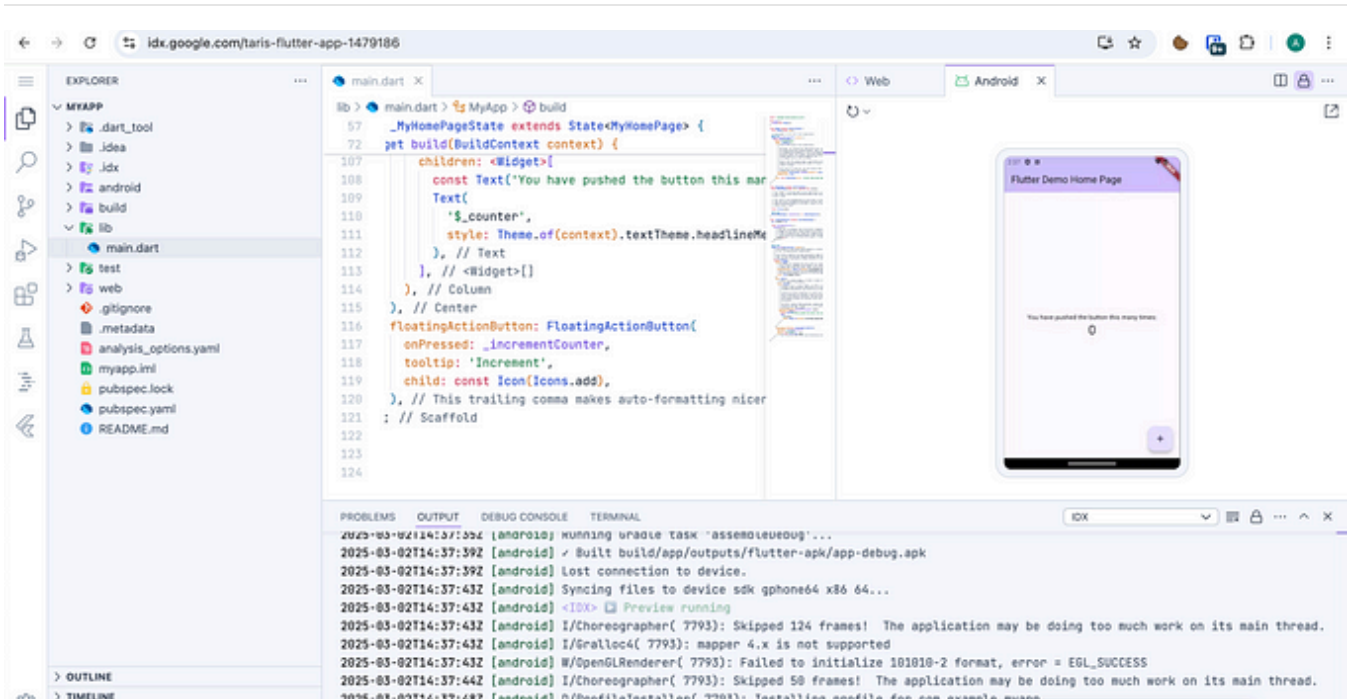
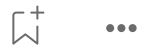



In ASP DOTNET by Engr. Md. Hasan Monsur

LINQ and Dapper in .NET: Best Practices for Efficient Data Queries

In this article, "LINQ and Dapper in .NET: Best Practices for Efficient Data Queries," we explore how to effectively leverage LINQ with...

✦ Oct 18, 2024 🖱️ 21

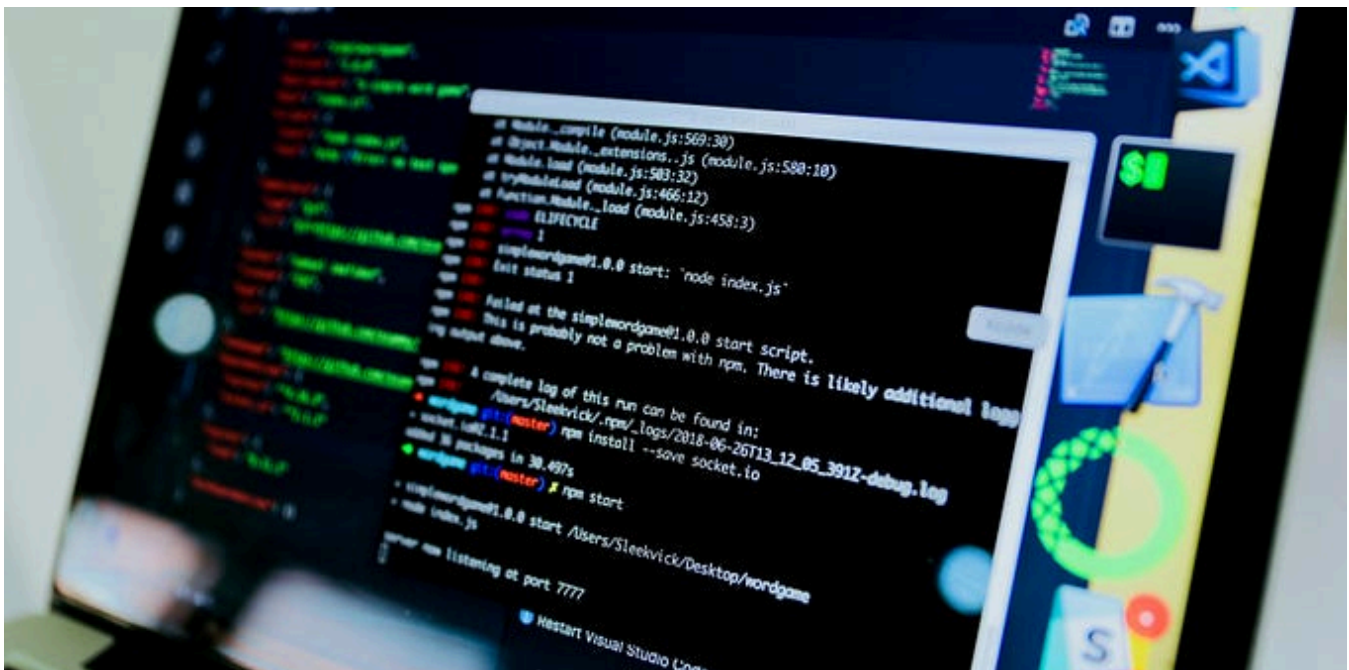


 In Coding Beauty by Tari Ibaba

This new IDE from Google is an absolute game changer

This new IDE from Google is seriously revolutionary.

✦ Mar 11 🖱️ 2.5K 💬 143





In ILLUMINATION by Piyush Kabra

From Development To Deployment: How DevOps Accelerates Time To Market

Docker and the Future of Deployment: How Containers Revolutionize Software Delivery



Jan 29



17



2



See more recommendations