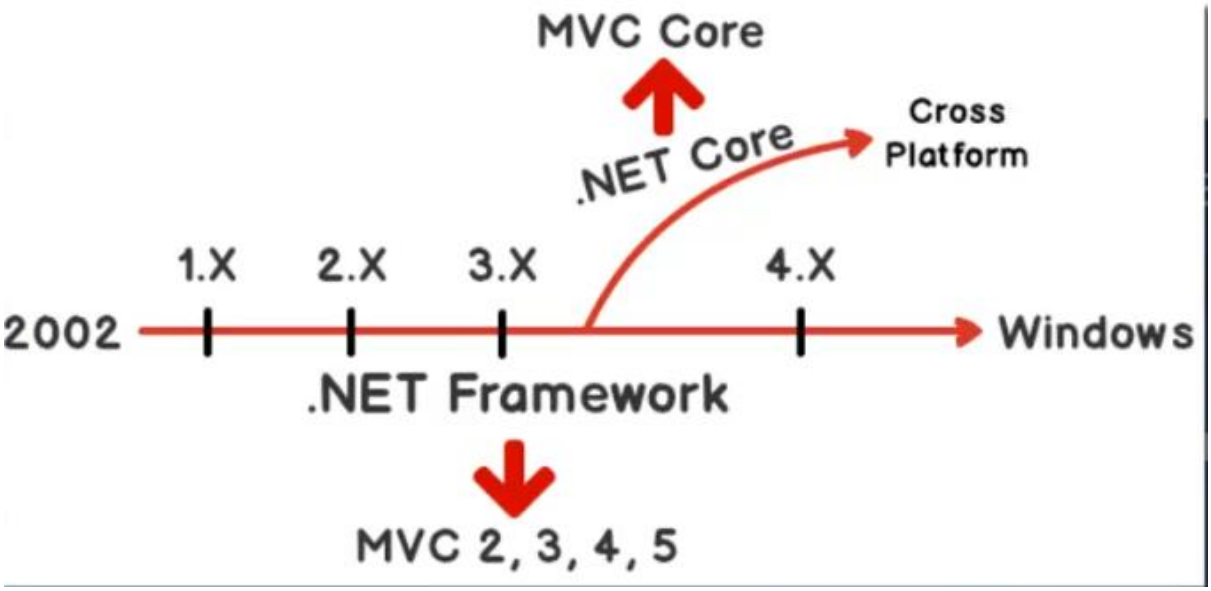
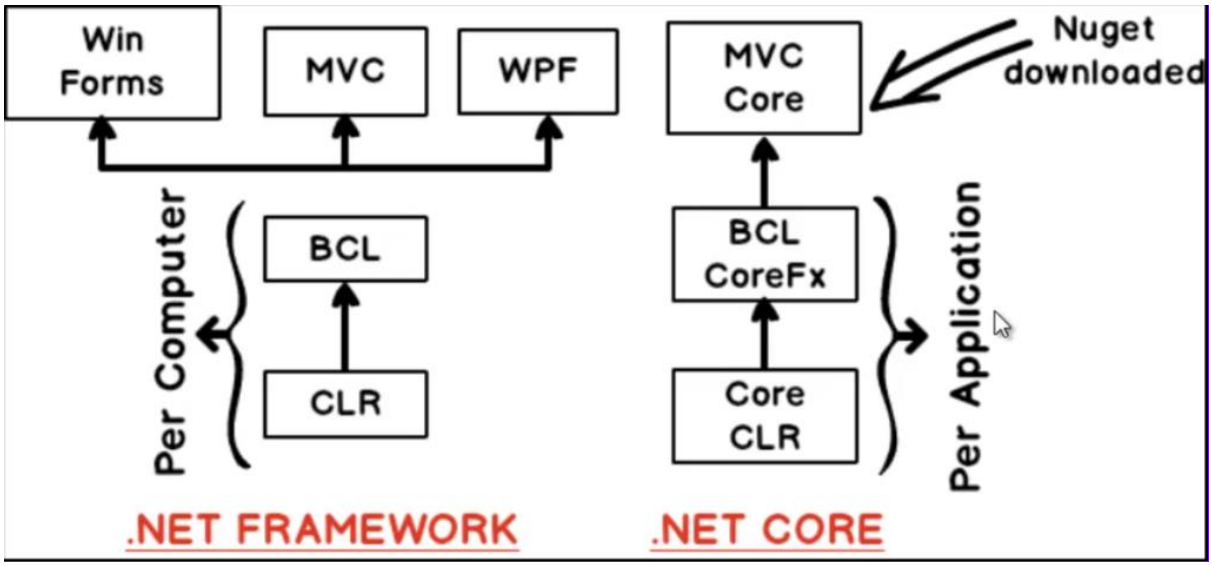
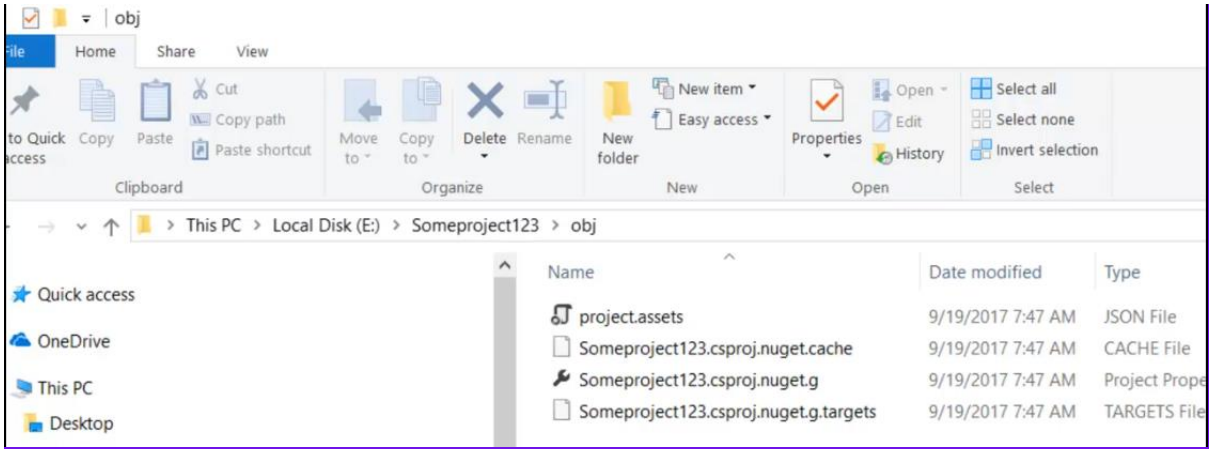
How is .NET Core different from .NET framework ?
What is Dotnet CLI ?
How to create a Dotnet Core application using .Net CLI ?
What is the need of PCL ?
What are the issues with PCL ?
Explain .NET Standards ?



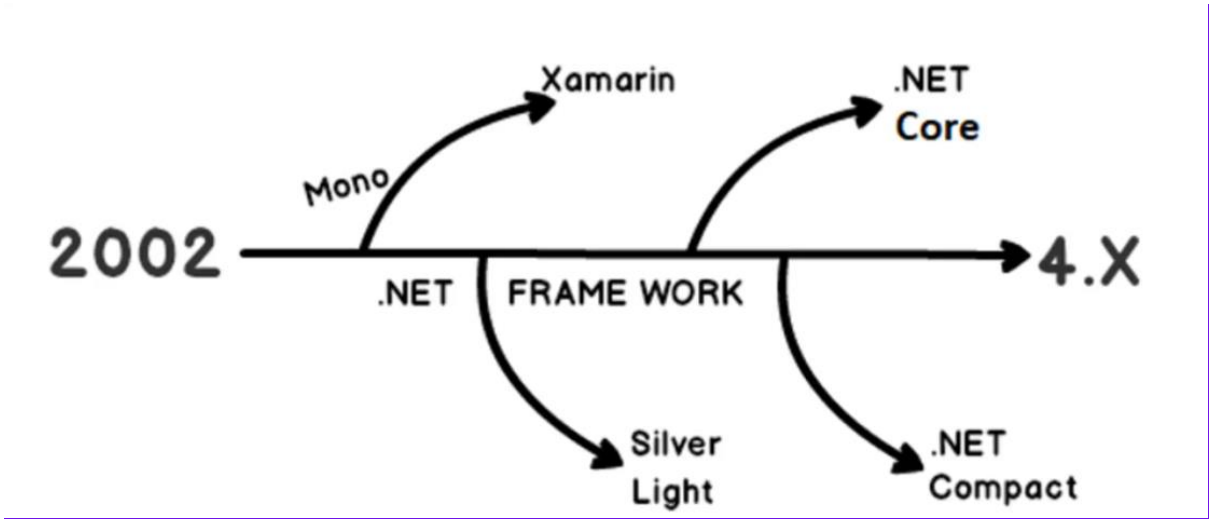
.NET Core is a cross-platform version of .NET for building websites, services, and console apps.

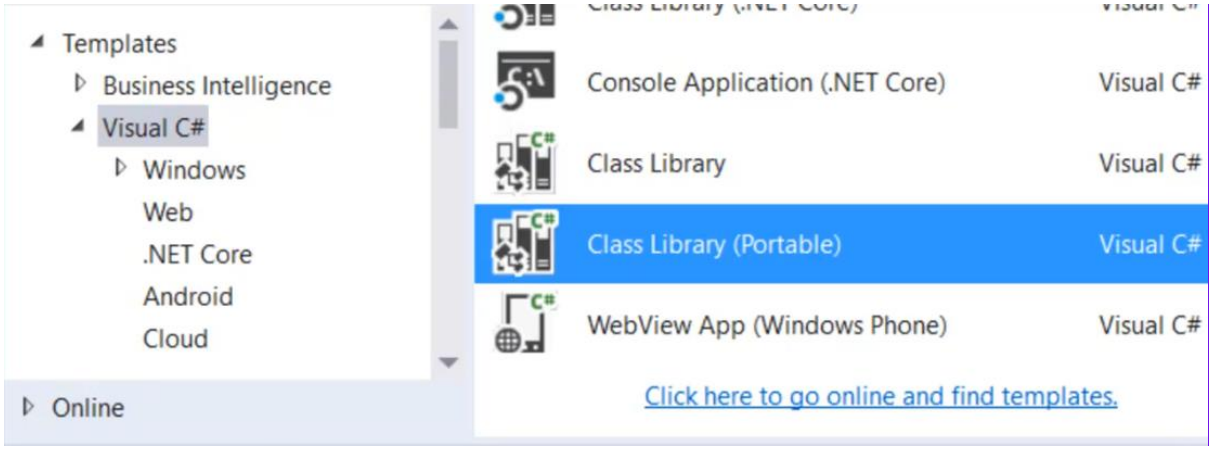


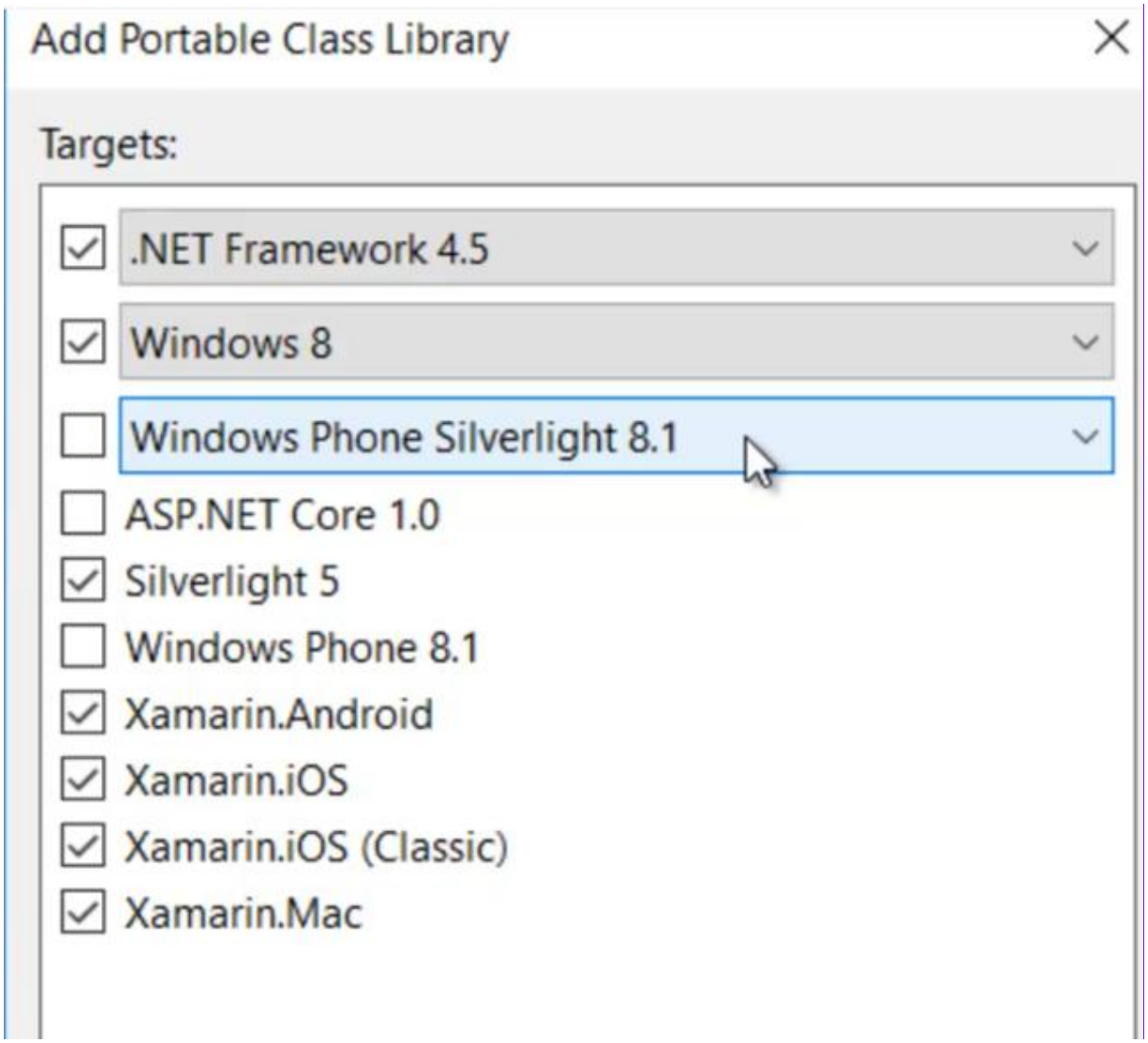
NuGet is the mechanism to bring in the component in your project.



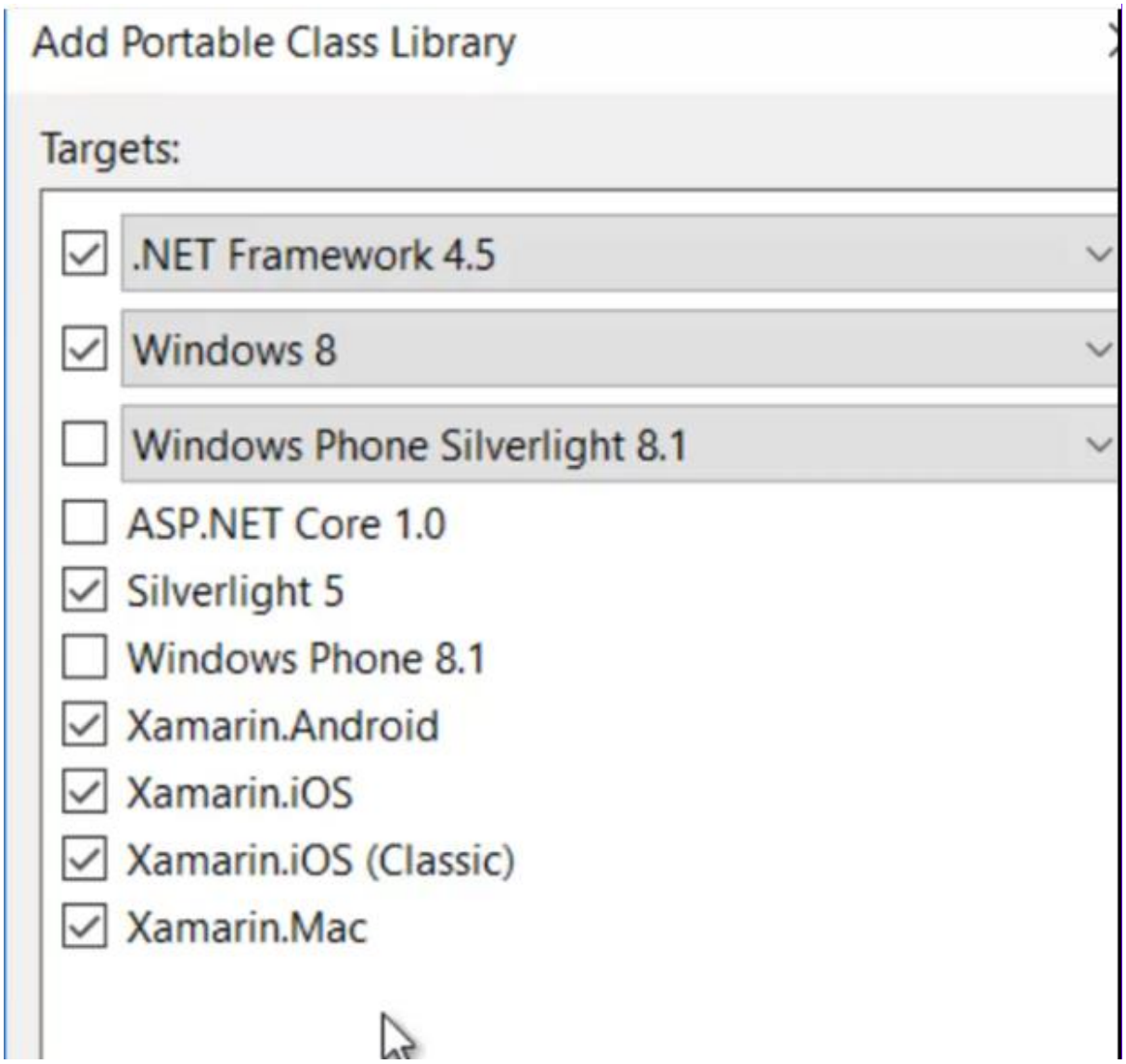

```
E:\Someproject123>dotnet new console
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on E:\Someproject123\Someproject123.csproj...
  Restoring packages for E:\Someproject123\Someproject123.csproj...
  Generating MSBuild file E:\Someproject123\obj\Someproject123.csproj.nuge
  Generating MSBuild file E:\Someproject123\obj\Someproject123.csproj.nuge
  Restore completed in 563.34 ms for E:\Someproject123\Someproject123.cspr


Restore succeeded.


E:\Someproject123>dotnet build
Microsoft (R) Build Engine version 15.3.409.57025 for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

  Someproject123 -> E:\Someproject123\bin\Debug\netcoreapp2.0\Someproject1

Build succeeded.
    0 Warning(s)
```

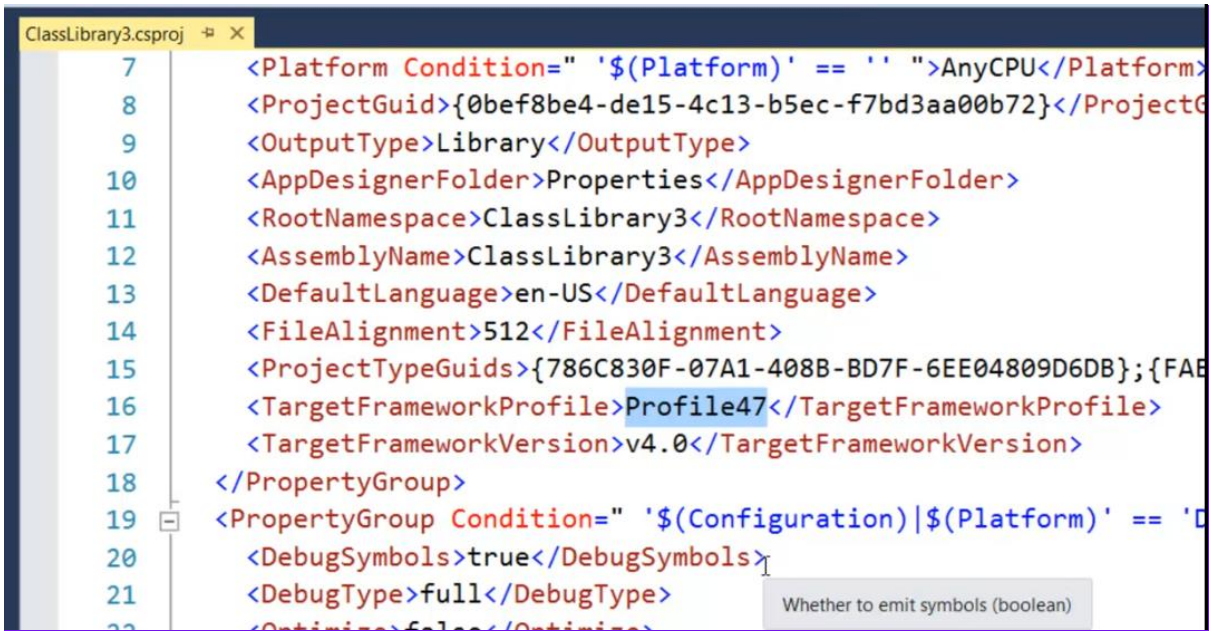






If suppose new family of .net introduced , then I have to again re compile my project


Add Portable Class Library
Targets:
.NET Framework 4.5
Windows 8
Windows Phone Silverlight 8.1
ASP.NET Core 1.0
Silverlight 5
Windows Phone 8.1
Xamarin.Android
Xamarin.iOS
Xamarin.iOS (Classic)
Xamarin.Mac


ClassLibrary3.csproj
<Platform Condition=" '$(Platform)' == '' ">AnyCPU</Platform>
<ProjectGuid>{0bef8be4-de15-4c13-b5ec-f7bd3aa00b72}</ProjectG
<OutputType>Library</OutputType>
<AppDesignerFolder>Properties</AppDesignerFolder>
<RootNamespace>ClassLibrary3</RootNamespace>
<AssemblyName>ClassLibrary3</AssemblyName>
<DefaultLanguage>en-US</DefaultLanguage>
<FileAlignment>512</FileAlignment>
<ProjectTypeGuids>{786C830F-07A1-408B-BD7F-6EE04809D6DB};{FAE
<TargetFrameworkProfile>Profile47</TargetFrameworkProfile>
<TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'D
<DebugSymbols>true</DebugSymbols>
<DebugType>full</DebugType>
Whether to emit symbols (boolean)

Portable Class Library (PCL) profiles as of **VS2015 Update 3**.

Filter

Include legacy portable profiles

Show full framework names

#	Frameworks	NuGet Target
Profile5	.NET Framework 4.0 Windows 8.0	portable-net4+win8
Profile6	.NET Framework 4.0.3 Windows 8.0	portable-net403+win8
Profile7	.NET Framework 4.5 Windows 8.0	portable-net45+win8
Profile14	.NET Framework 4.0	portable-net4+sl50

platforms you intend to run on. For insta

target .NET Standard 1.1.

The .NET Standard is a formal specification of .NET APIs that are intended to be available on all .NET implementations. The motivation behind the .NET Standard is establishing greater uniformity in the .NET ecosystem

.NET Standard	1.0	1.1	1.2	1.3	1.4	1.5	
.NET Core	1.0	1.0	1.0	1.0	1.0	1.0	
.NET Framework	4.5	4.5	4.5.1	4.6	4.6.1	4.6.1 ~~4.6.2~~	4.6.1
Mono	4.6	4.6	4.6	4.6	4.6	4.6	
Xamarin.iOS	10.0	10.0	10.0	10.0	10.0	10.0	
Xamarin.Mac	3.0	3.0	3.0	3.0	3.0	3.0	
Xamarin.Android	7.0	7.0	7.0	7.0	7.0	7.0	
Universal Windows Platform	10.0	10.0	10.0	10.0	10.0	vNext	

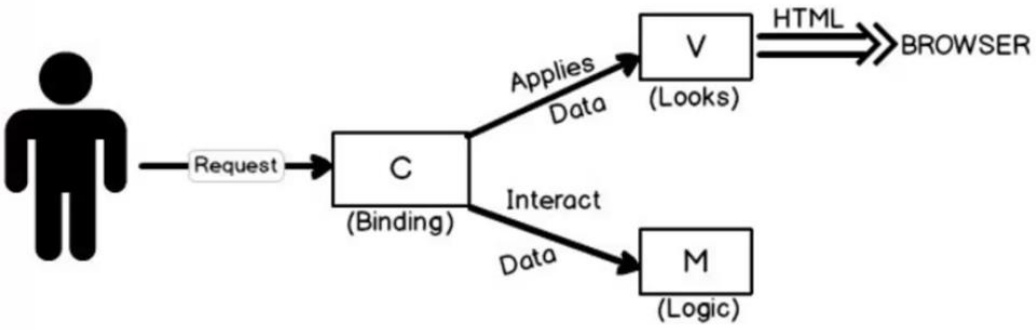
platforms you intend to run on. For i 5 and .N
target .NET Standard 1.1.

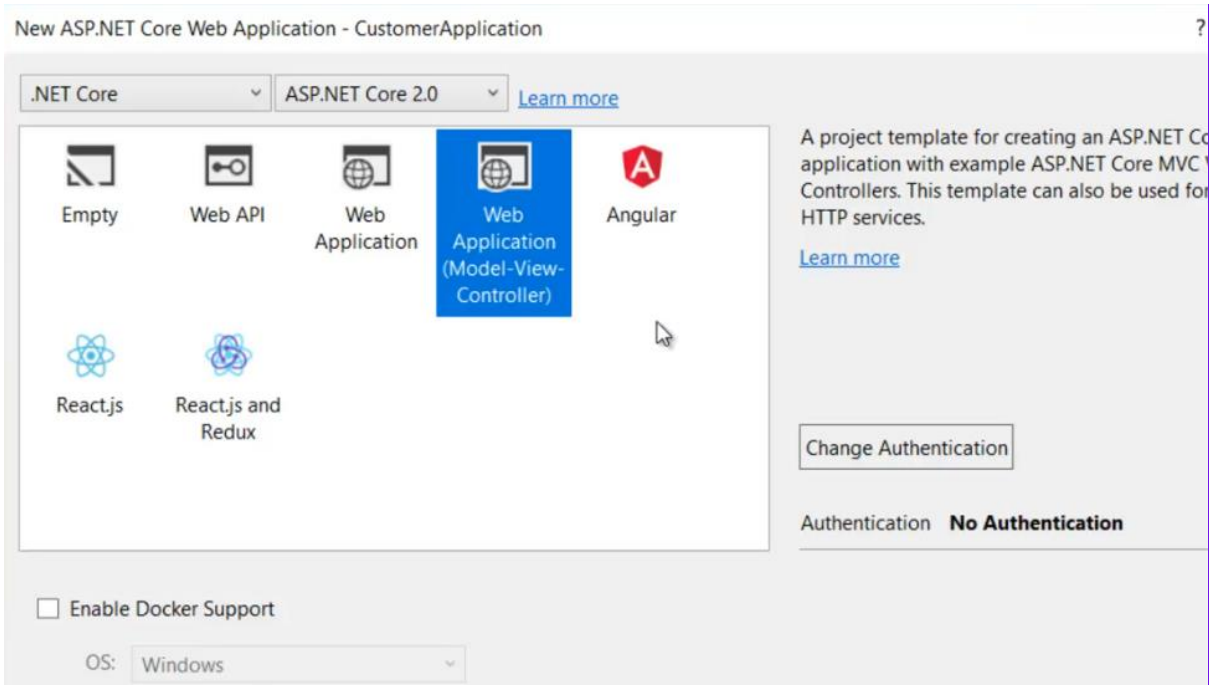
The problem is centralized control of standards.

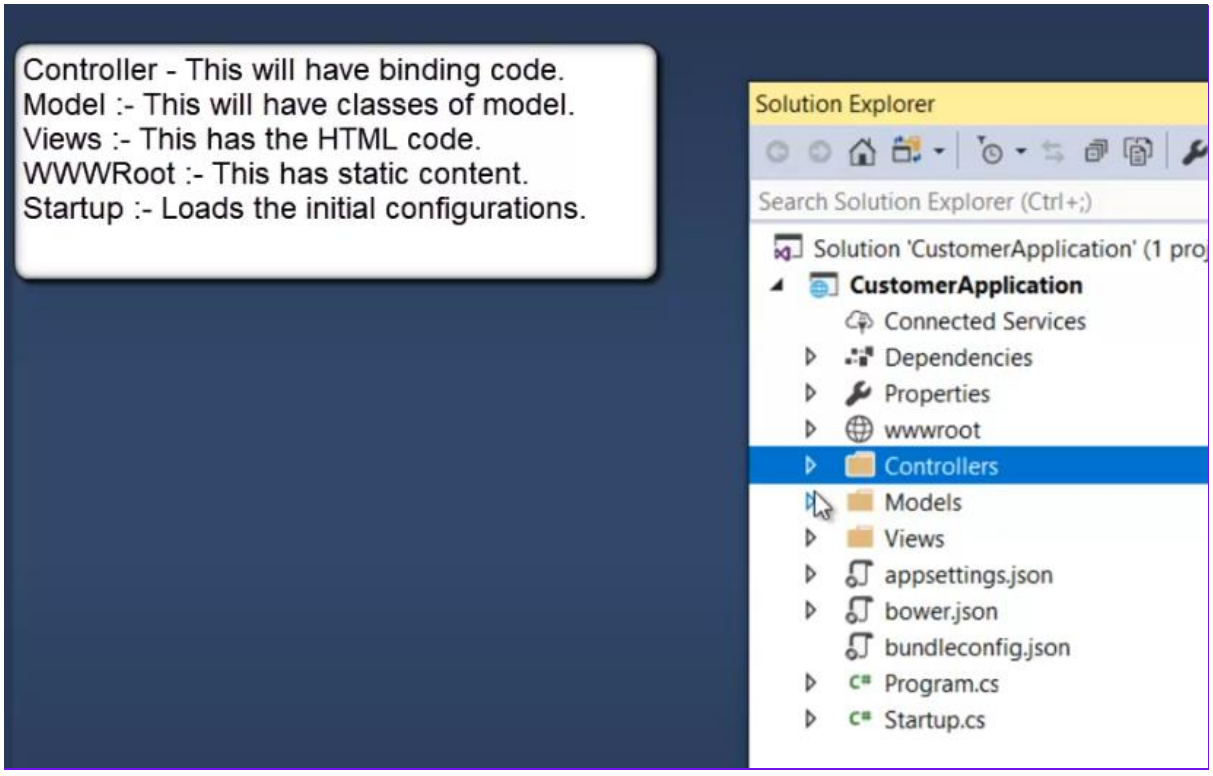
.NET Standard	1.0	1.1	1.2	1.3	1.4	1.5	
.NET Core	1.0	1.0	1.0	1.0	1.0	1.0	
.NET Framework	4.5	4.5	4.5.1	4.6	4.6.1	4.6.1 ~~4.6.2~~	4.6.1
Mono	4.6	4.6	4.6	4.6	4.6	4.6	
Xamarin.iOS	10.0	10.0	10.0	10.0	10.0	10.0	
Xamarin.Mac	3.0	3.0	3.0	3.0	3.0	3.0	
Xamarin.Android	7.0	7.0	7.0	7.0	7.0	7.0	
Universal Windows Platform	10.0	10.0	10.0	10.0	10.0	vNext	

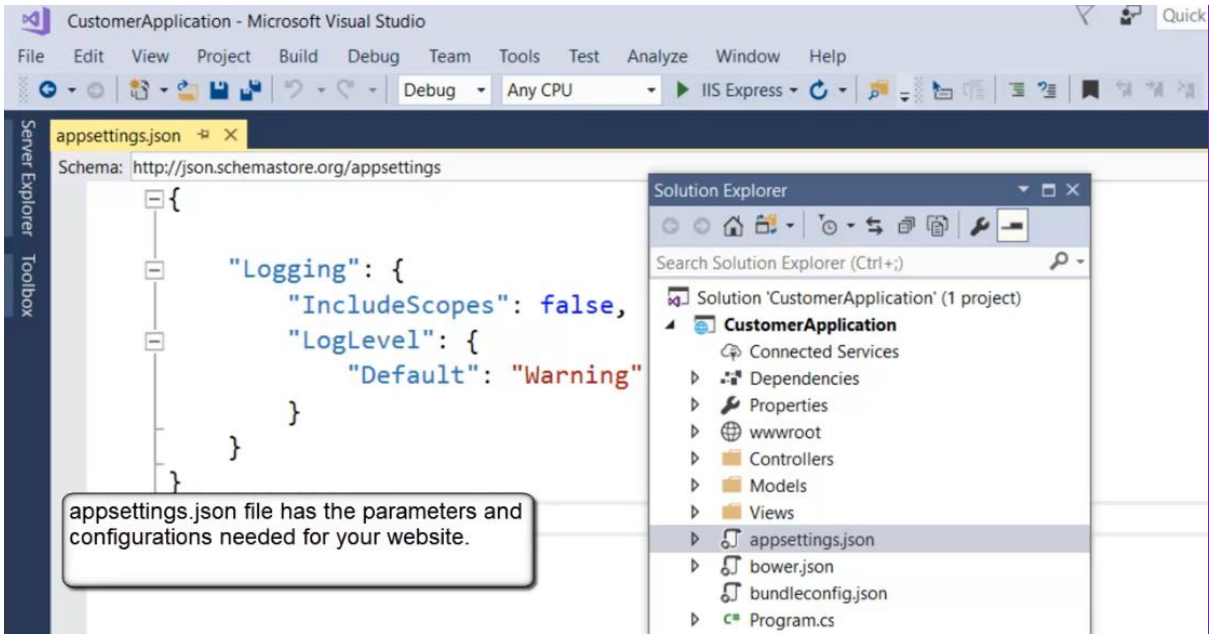
Understanding MVC architecture.
Creating a MVC core project.
Need of Program.cs and startup.cs file.
Importance of appsettings.json file.
How to read configuration data ?.
Kestrel webserver and reverse proxy concept.
Configuration and Configuration Services.
Revising the full flow of MVC startup proces

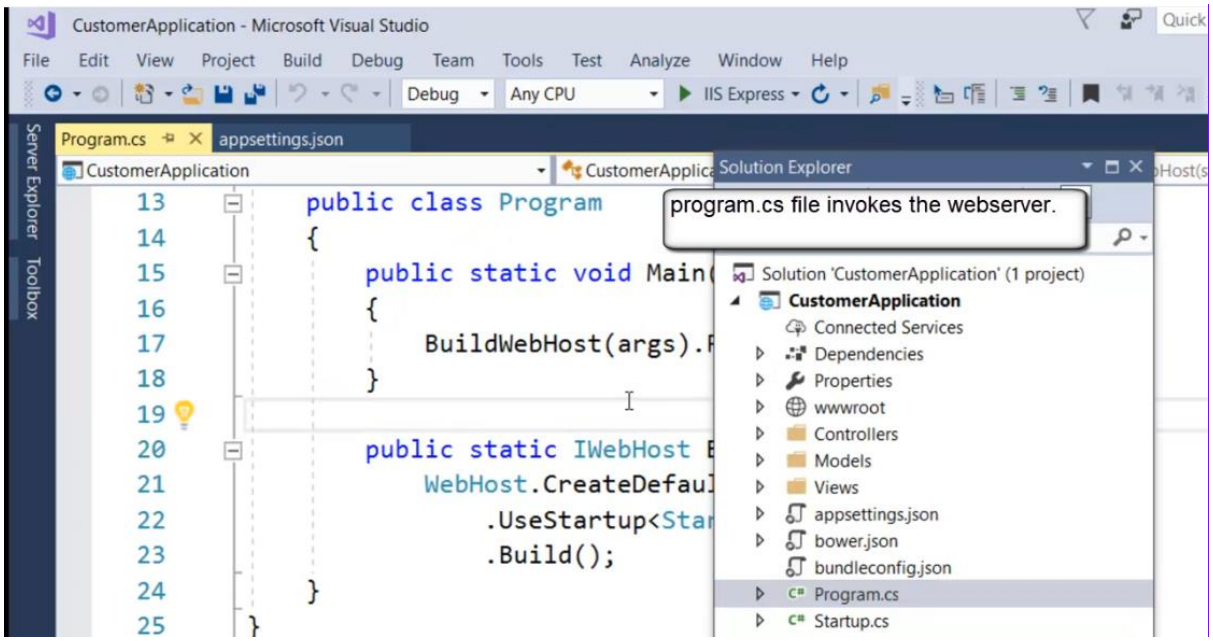
C :- Controller , M :- Model , V :- View

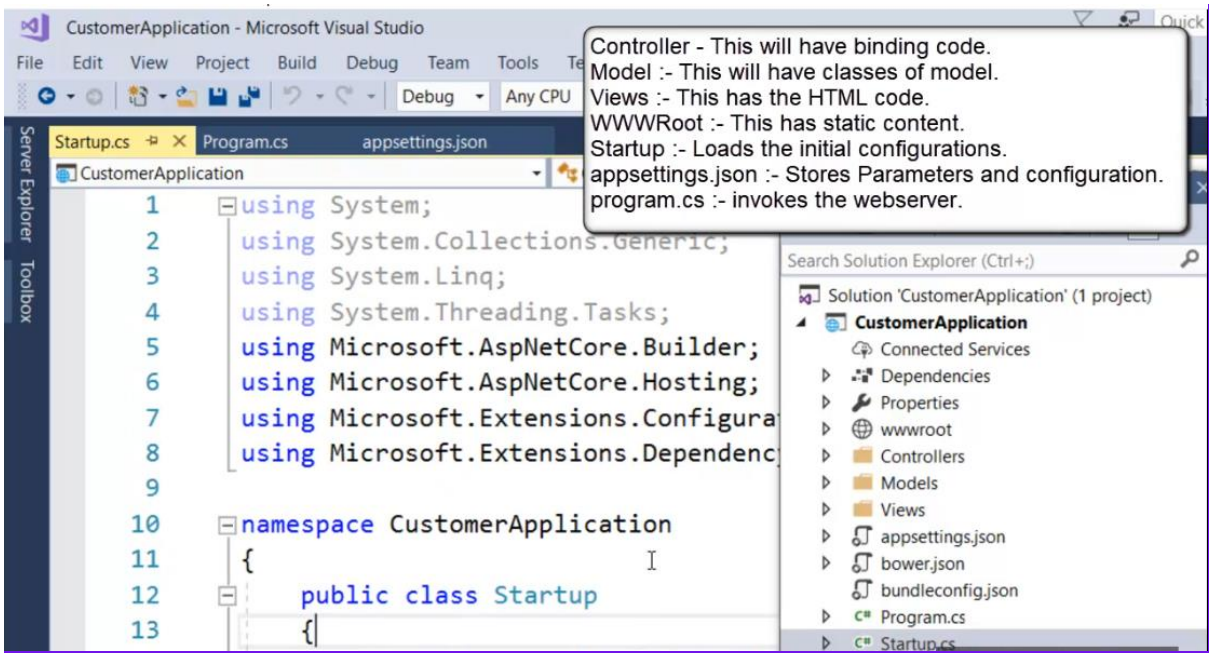


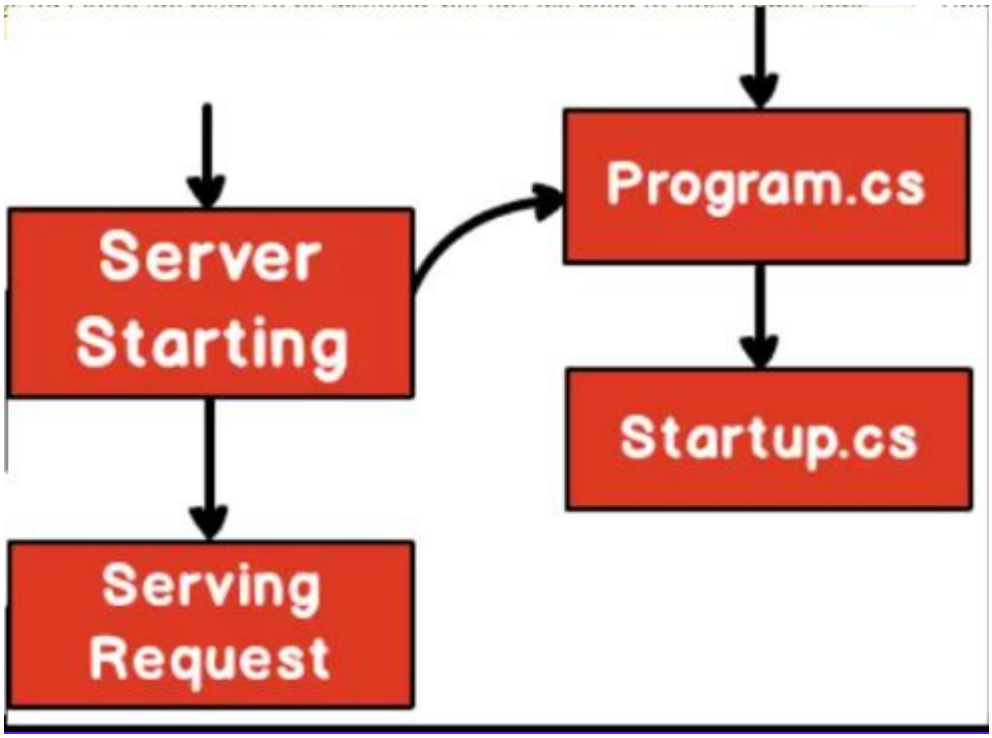

New ASP.NET Core Web Application - CustomerApplication
.NET Core
ASP.NET Core 2.0
Learn more
Empty
Web API
Web Application
Web Application (Model-View-Controller)
Angular
React.js
React.js and Redux
A project template for creating an ASP.NET Co application with example ASP.NET Core MVC Controllers. This template can also be used for HTTP services.
Learn more
Change Authentication
Authentication No Authentication
Enable Docker Support
OS: Windows

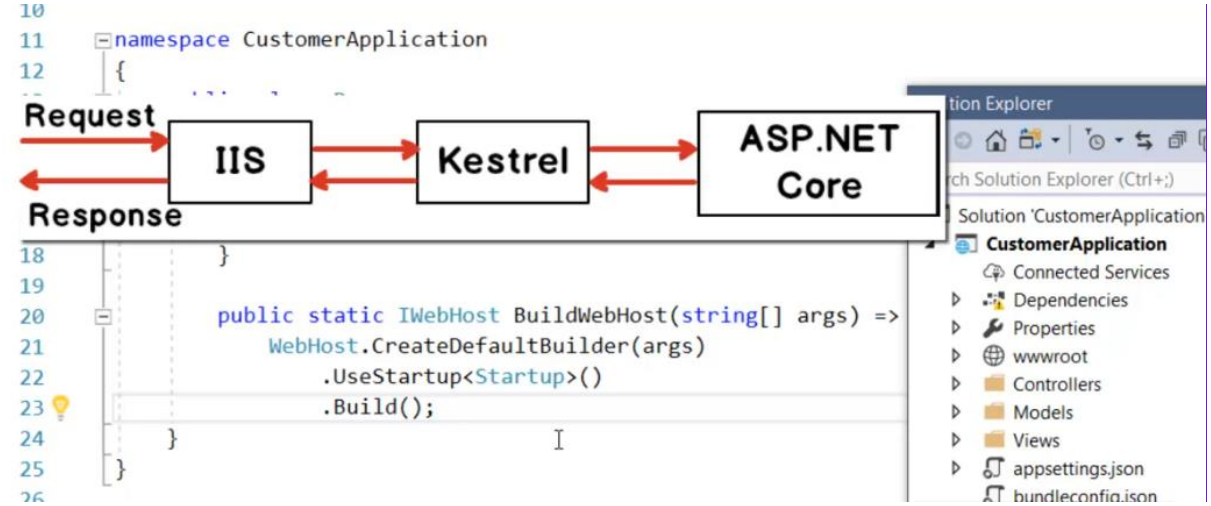

Controller - This will have binding code.
Model :- This will have classes of model.
Views :- This has the HTML code.
WWWRoot :- This has static content.
Startup :- Loads the initial configurations.
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'CustomerApplication' (1 proj
CustomerApplication
Connected Services
Dependencies
Properties
wwwroot
Controllers
Models
Views
appsettings.json
bower.json
bundleconfig.json
Program.cs
Startup.cs

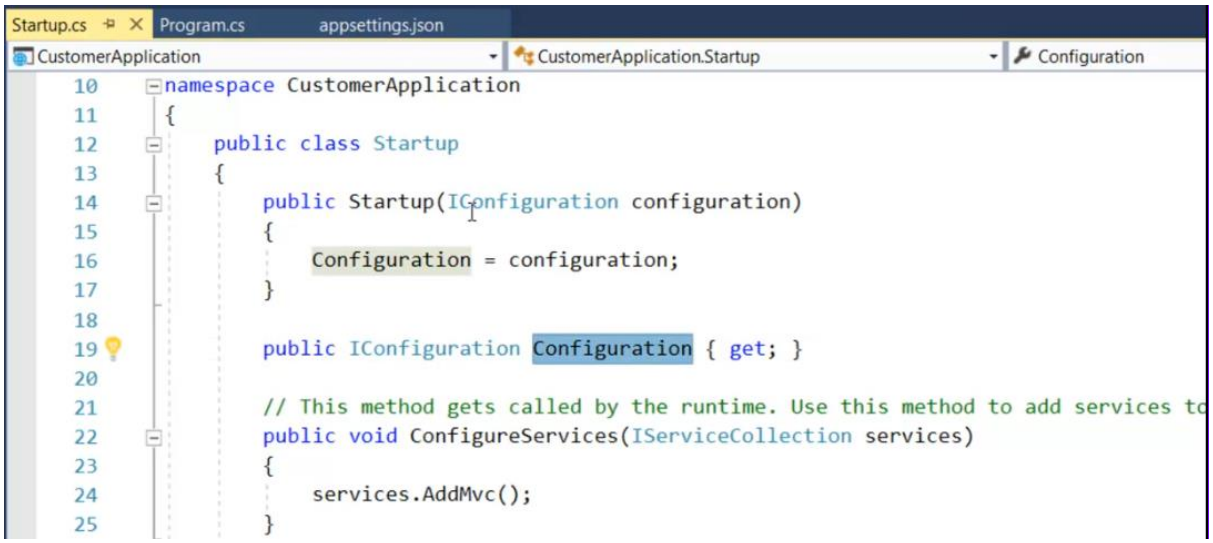

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
appsettings.json
Schema: http://json.schemastore.org/appsettings
{
"Logging": {
"IncludeScopes": false,
"LogLevel": {
"Default": "Warning"
}
}
}
appsettings.json file has the parameters and configurations needed for your website.
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'CustomerApplication' (1 project)
CustomerApplication
Connected Services
Dependencies
Properties
wwwroot
Controllers
Models
Views
appsettings.json
bower.json
bundleconfig.json
Program.cs

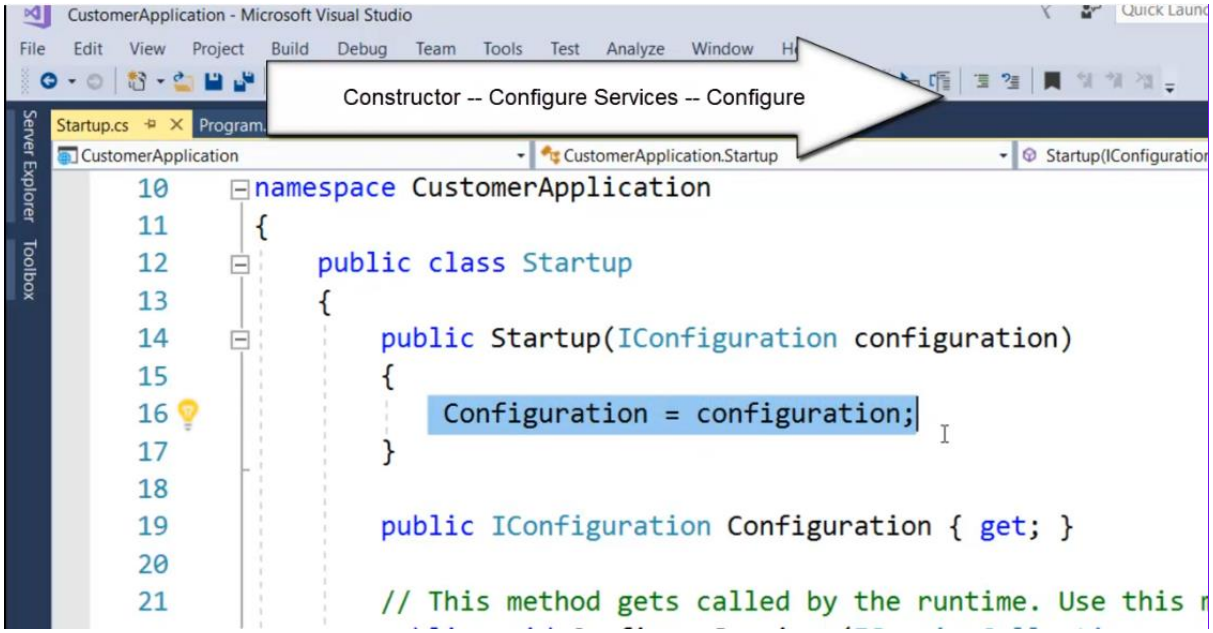

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Program.cs
appsettings.json
CustomerApplication
public class Program
{
public static void Main(
{
BuildWebHost(args).R
}
public static IWebHost B
WebHost.CreateDefaul
.UseStartup<Star
.Build();
}
}
program.cs file invokes the webserver.
Solution Explorer
Solution 'CustomerApplication' (1 project)
CustomerApplication
Connected Services
Dependencies
Properties
wwwroot
Controllers
Models
Views
appsettings.json
bower.json
bundleconfig.json
Program.cs
Startup.cs


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools
Debug
Any CPU
Startup.cs
Program.cs
appsettings.json
CustomerApplication
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5 using Microsoft.AspNetCore.Builder;
6 using Microsoft.AspNetCore.Hosting;
7 using Microsoft.Extensions.Configura
8 using Microsoft.Extensions.Dependenc
9
10 namespace CustomerApplication
11 {
12 public class Startup
13 {
Controller - This will have binding code.
Model :- This will have classes of model.
Views :- This has the HTML code.
WWWRoot :- This has static content.
Startup :- Loads the initial configurations.
appsettings.json :- Stores Parameters and configuration.
program.cs :- invokes the webserver.
Search Solution Explorer (Ctrl+;)
Solution 'CustomerApplication' (1 project)
CustomerApplication
Connected Services
Dependencies
Properties
wwwroot
Controllers
Models
Views
appsettings.json
bower.json
bundleconfig.json
Program.cs
Startup.cs
Server Explorer
Toolbox

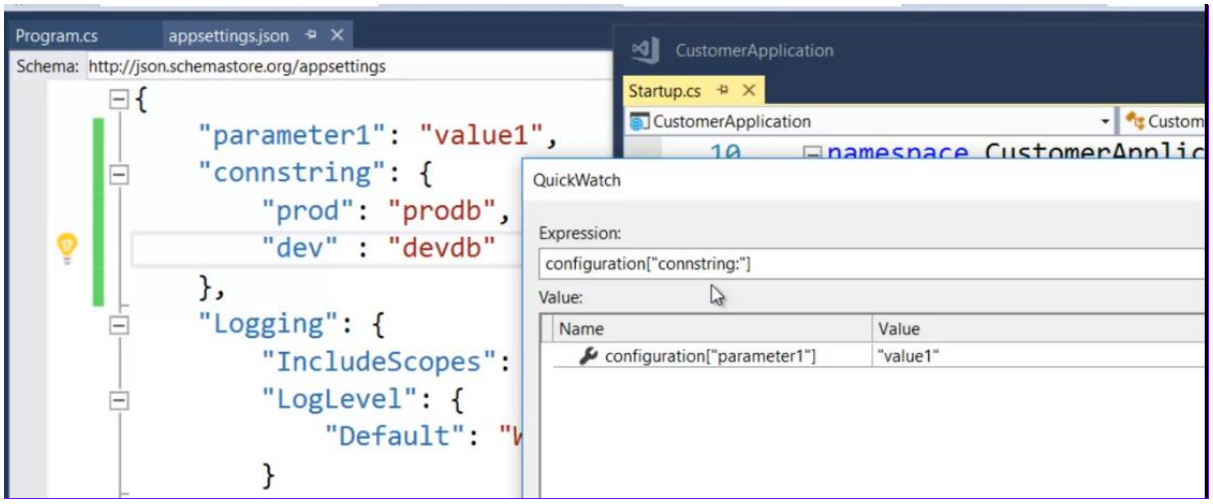

Server Starting
Program.cs
Startup.cs
Serving Request

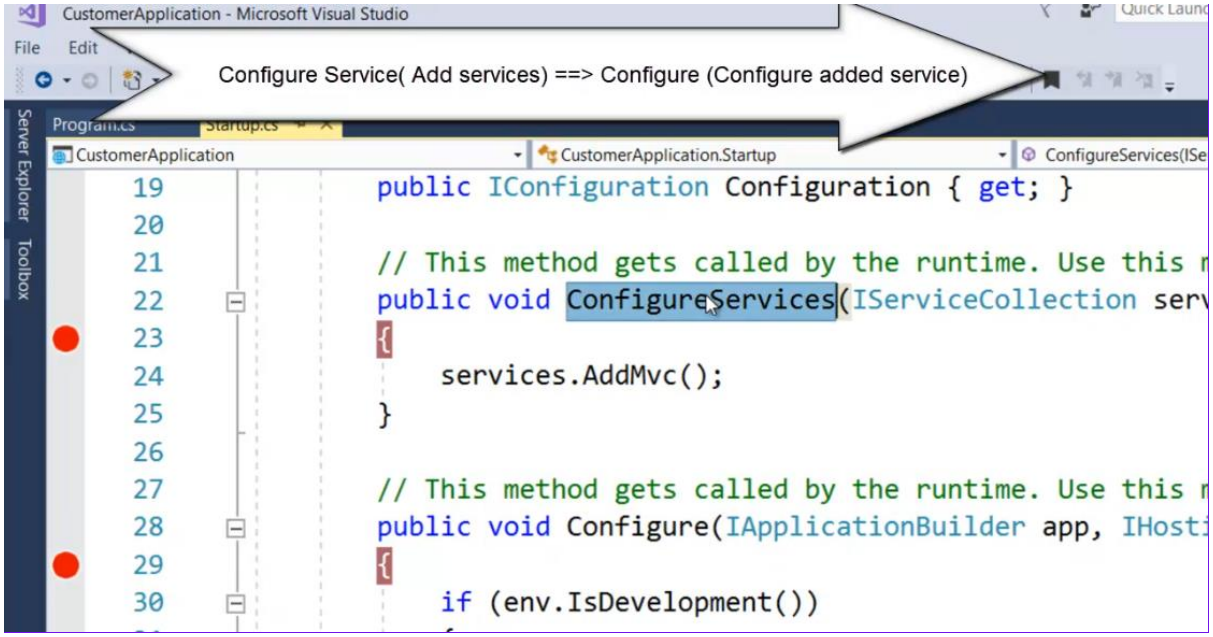




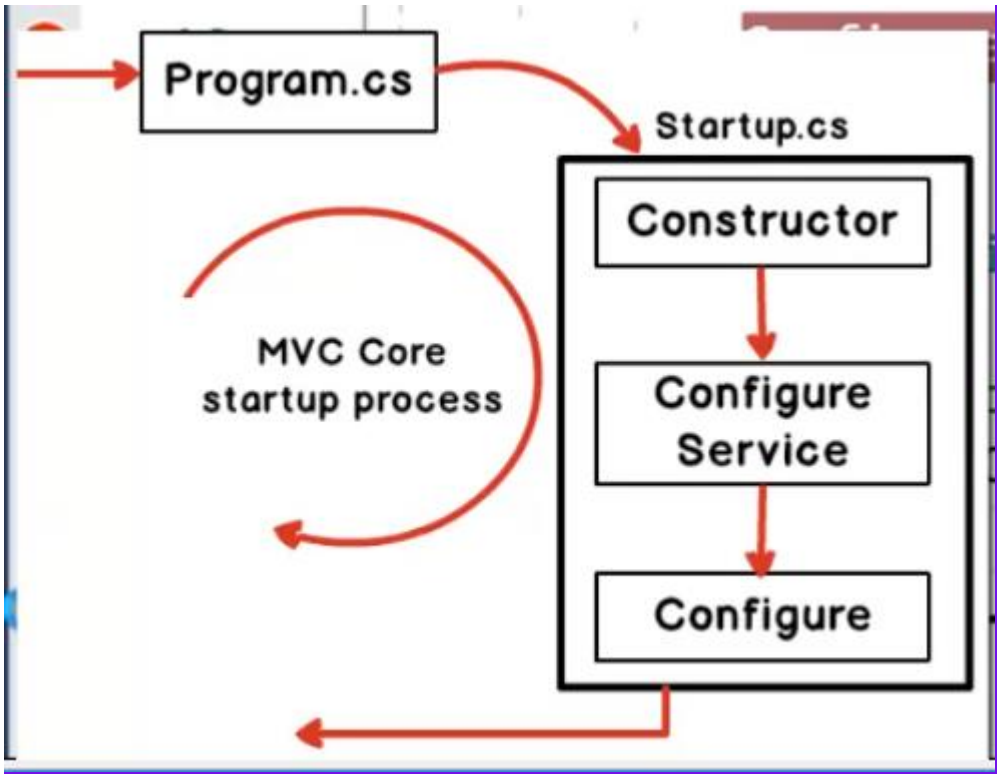


This configuration using dependency injection.

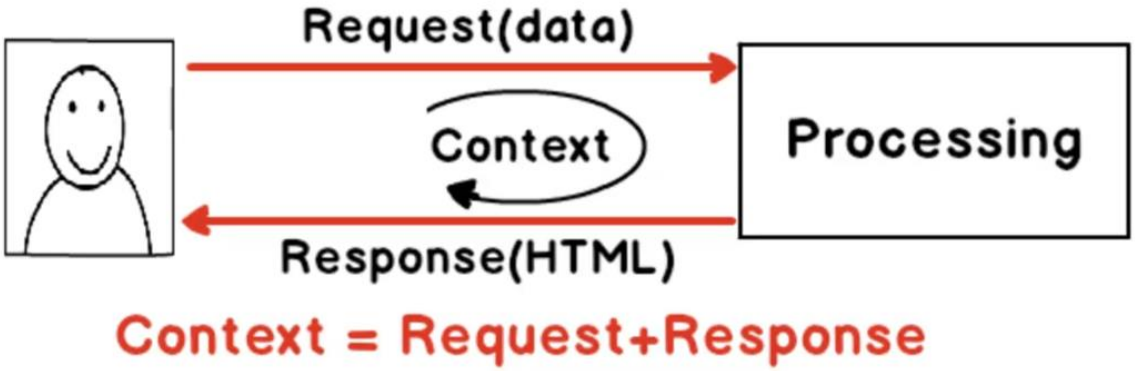


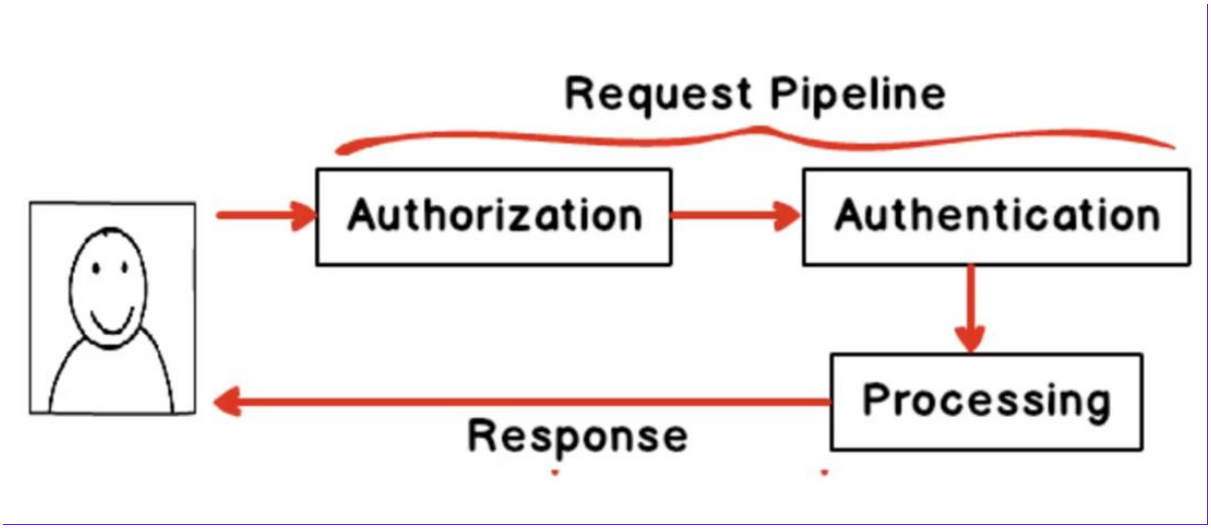


These two method are used to see , which service you want to use and how you want to configure them.

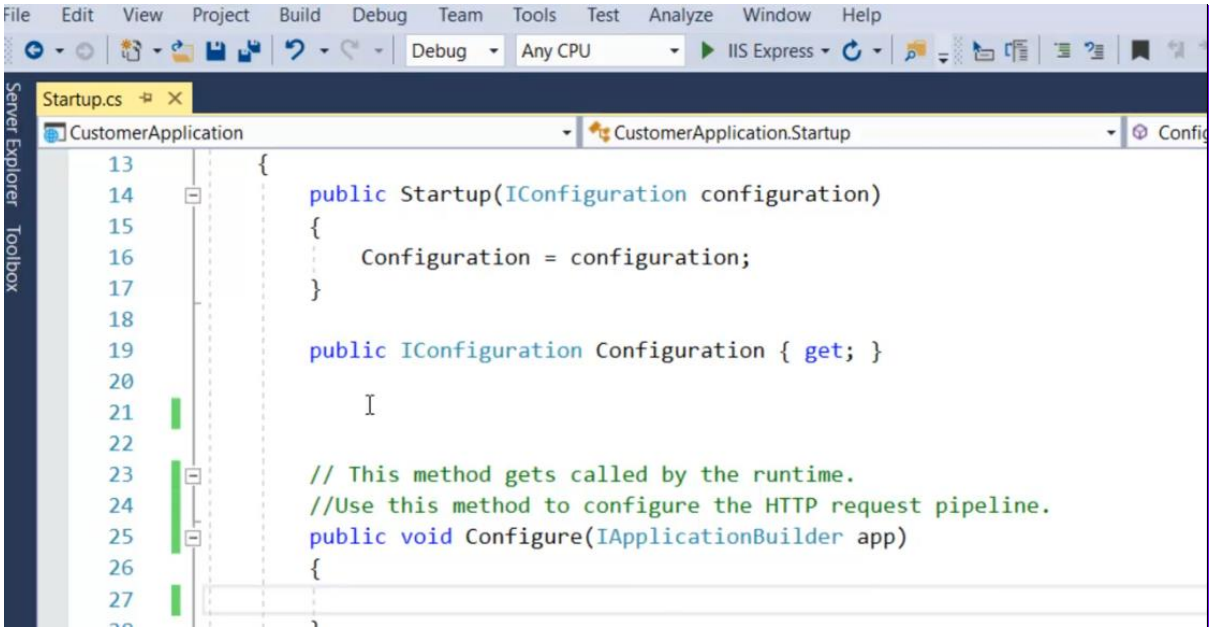


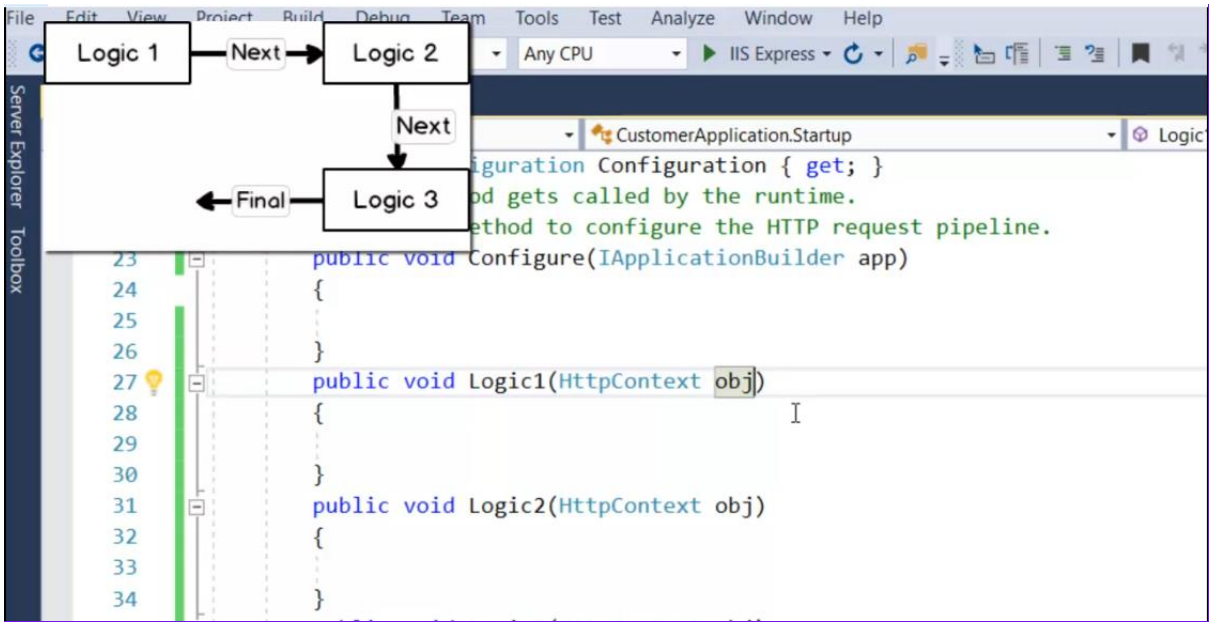
MVC they have made plug and play model

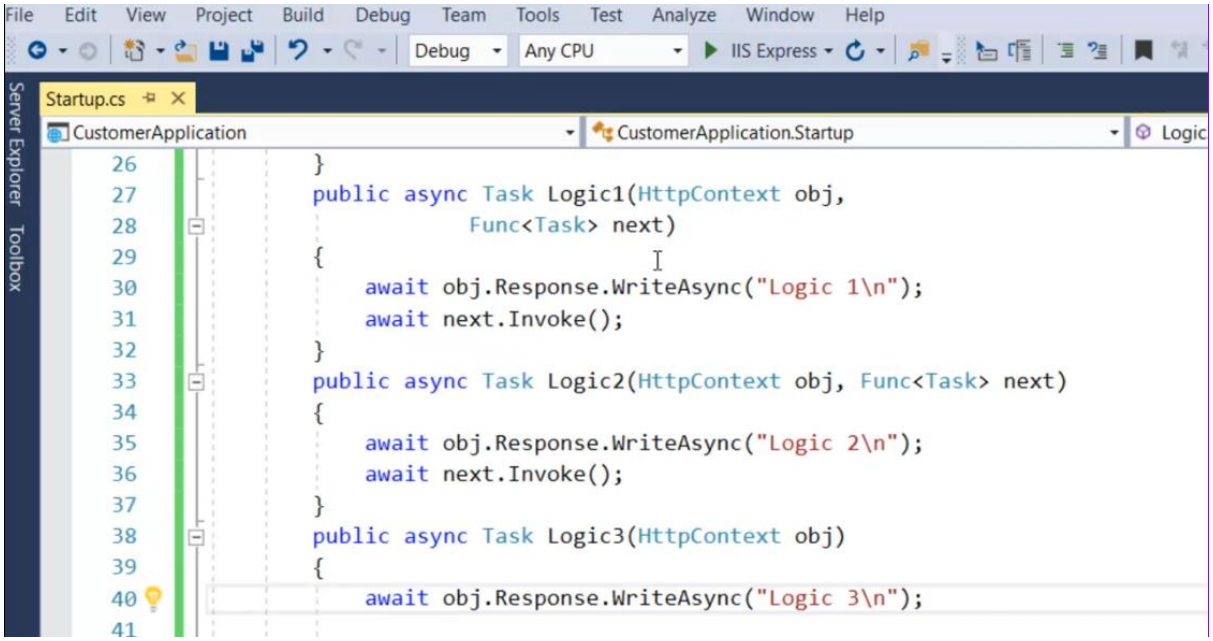


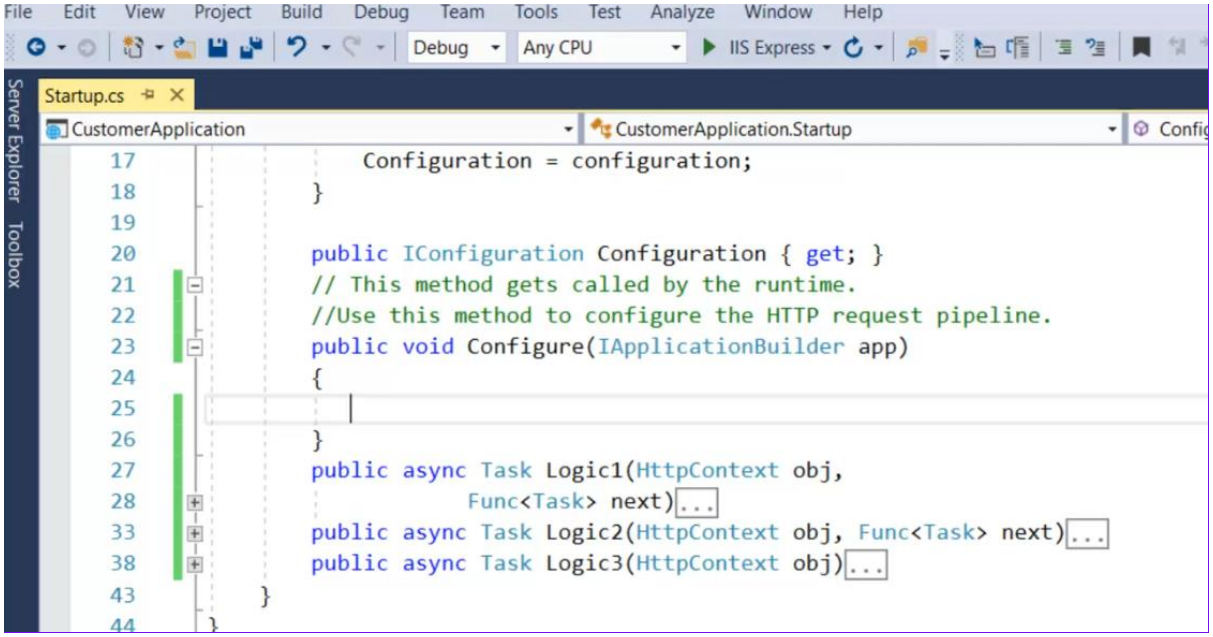

Request Pipeline
Authorization
Authentication
Processing
Response


Startup.cs
CustomerApplication
CustomerApplication.Startup
public void ConfigureServices(IServiceCollection services)
{
services.AddMvc();
}
// This method gets called by the runtime. Use this method
public void Configure(IApplicationBuilder app, IHostingEnvi
{
if (env.IsDevelopment())
{
app.UseDeveloperExceptionPage();
app.UseBrowserLink();
}
else
{
app.UseExceptionHandler("/Home/Error");
Solution Explo
Search Solutio
Solution

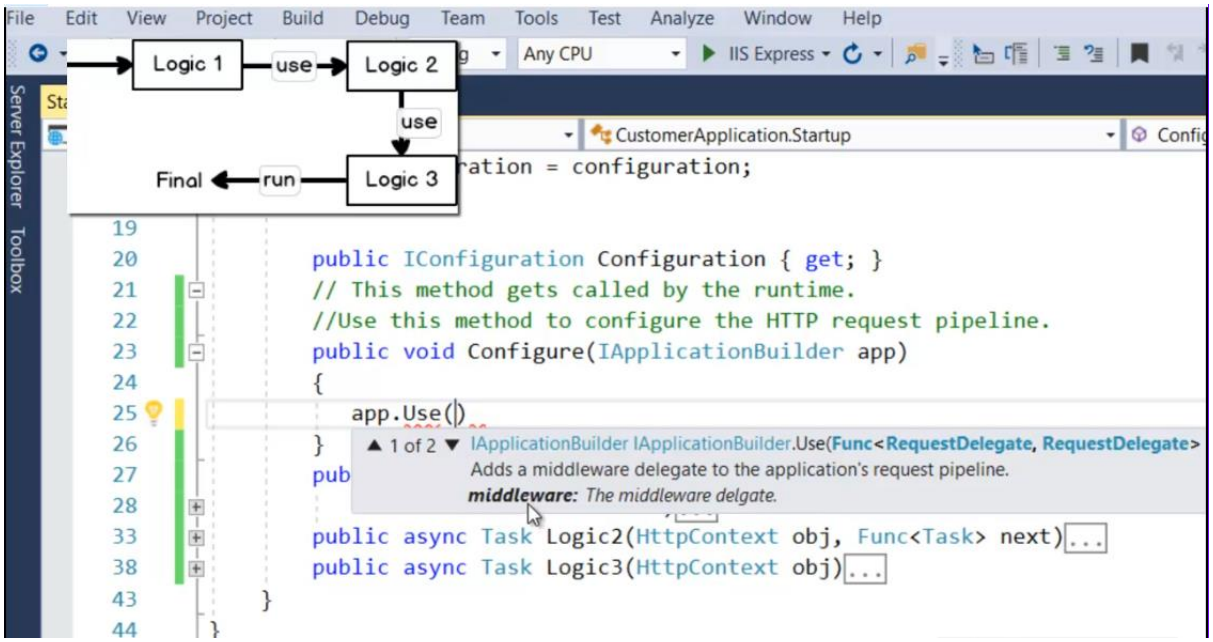

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Startup.cs
CustomerApplication
CustomerApplication.Startup
{
public Startup(IConfiguration configuration)
{
Configuration = configuration;
}
public IConfiguration Configuration { get; }
// This method gets called by the runtime.
//Use this method to configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app)
{


Logic 1
Next
Logic 2
Next
Final
Logic 3
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Any CPU
IIS Express
Server Explorer
Toolbox
CustomerApplication.Startup
iguration Configuration { get; }
od gets called by the runtime.
ethod to configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app)
{
}
public void Logic1(HttpContext obj)
{
}
public void Logic2(HttpContext obj)
{
}

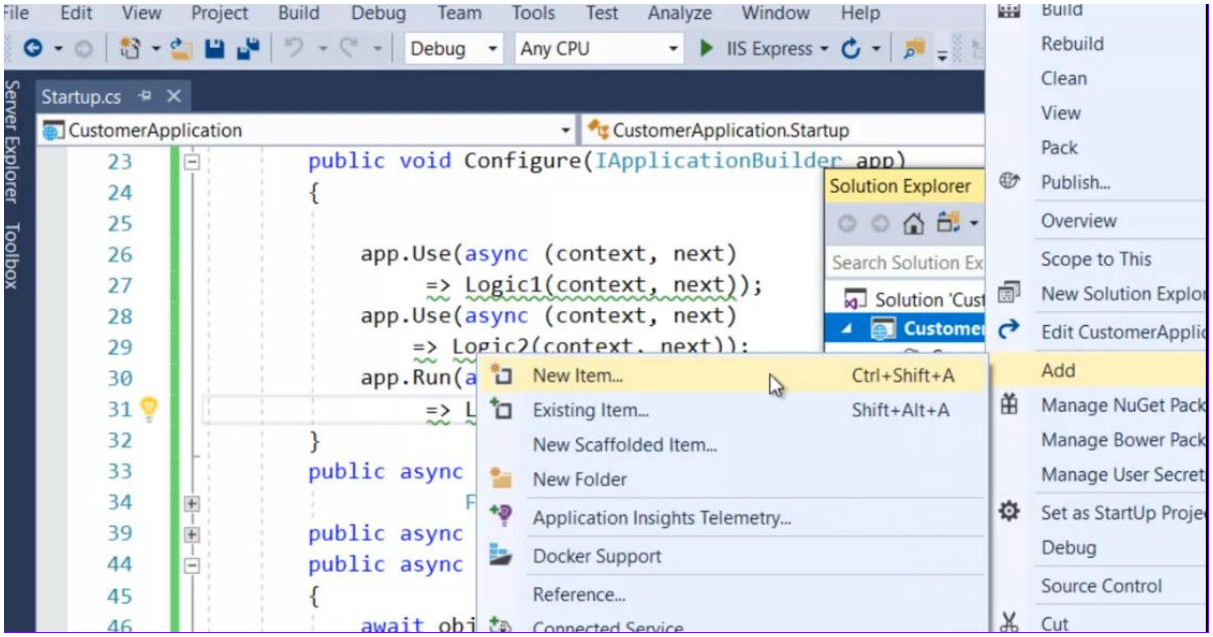

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Server Explorer
Toolbox
Startup.cs
CustomerApplication
CustomerApplication.Startup
}
public async Task Logic1(HttpContext obj,
Func<Task> next)
{
await obj.Response.WriteAsync("Logic 1\n");
await next.Invoke();
}
public async Task Logic2(HttpContext obj, Func<Task> next)
{
await obj.Response.WriteAsync("Logic 2\n");
await next.Invoke();
}
public async Task Logic3(HttpContext obj)
{
await obj.Response.WriteAsync("Logic 3\n");



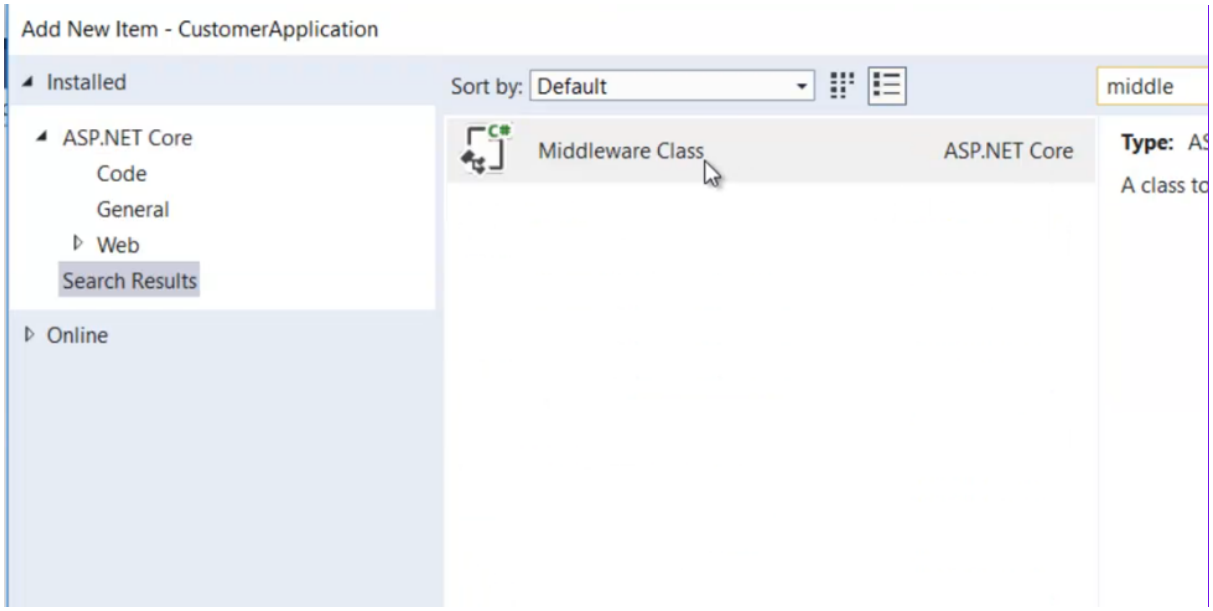
This app has two method USE and RUN



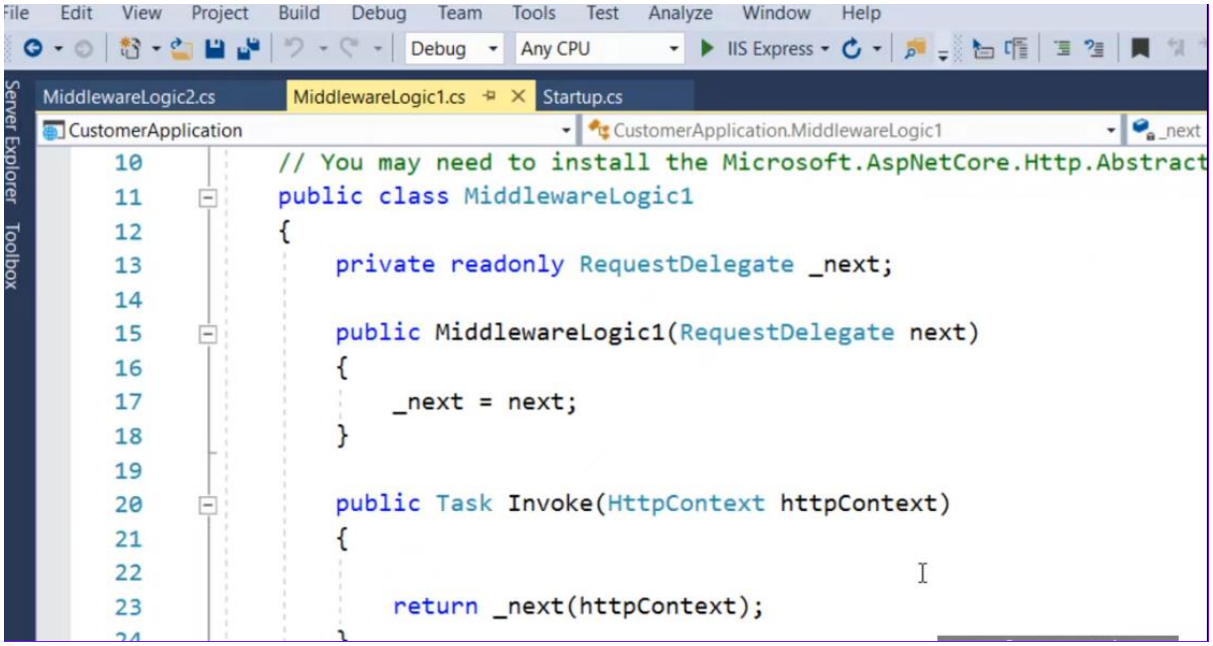
They are the middle ware

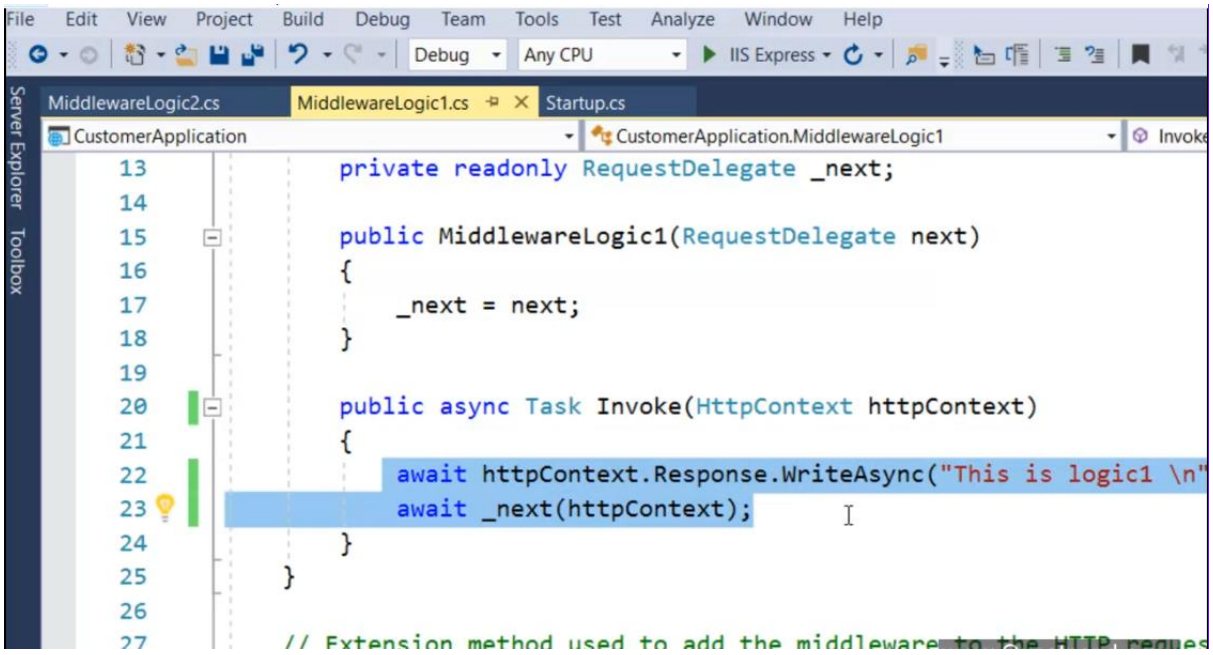


In order to add middle ware add class you can template also

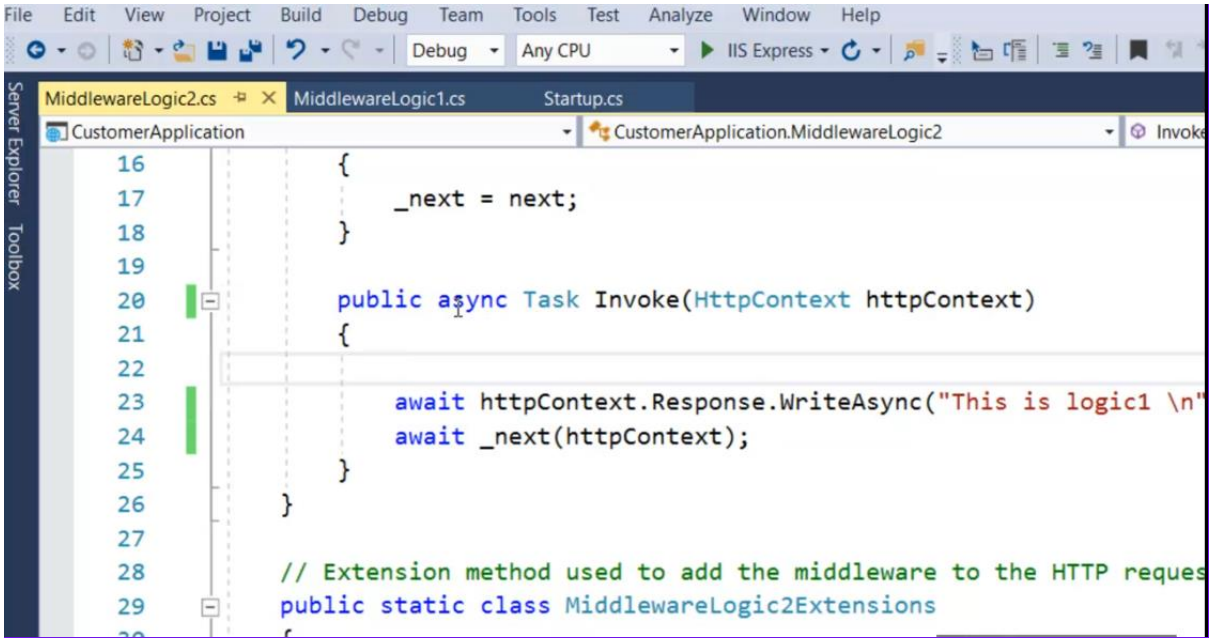


Add three two Middleware file

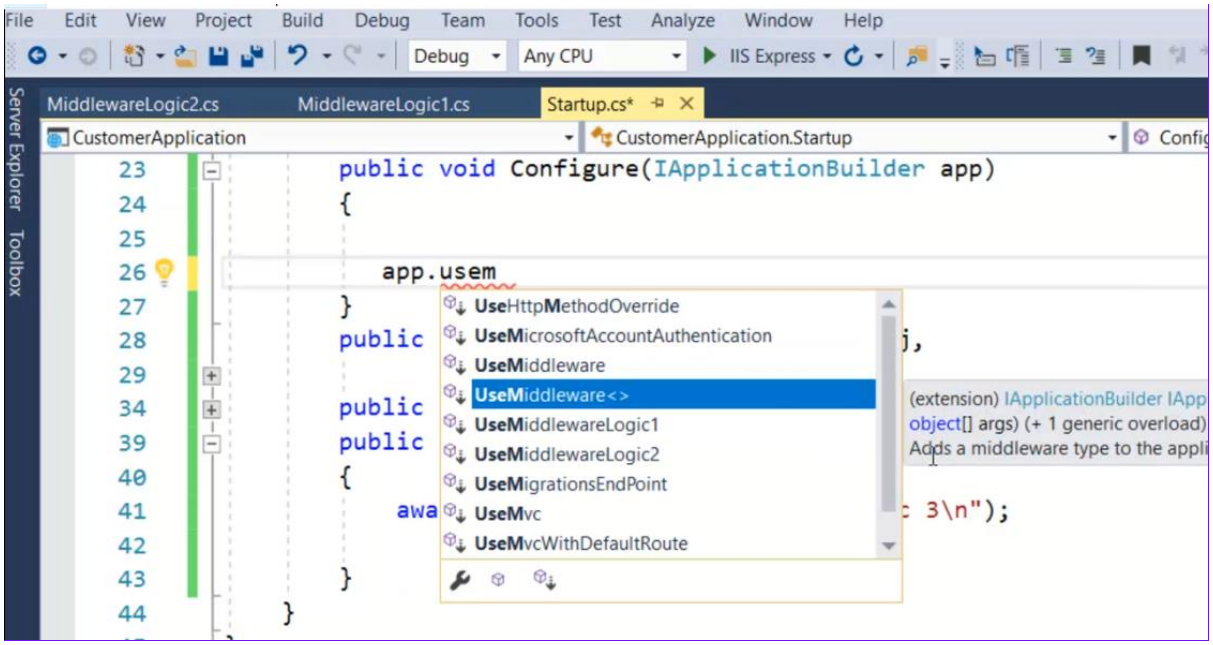


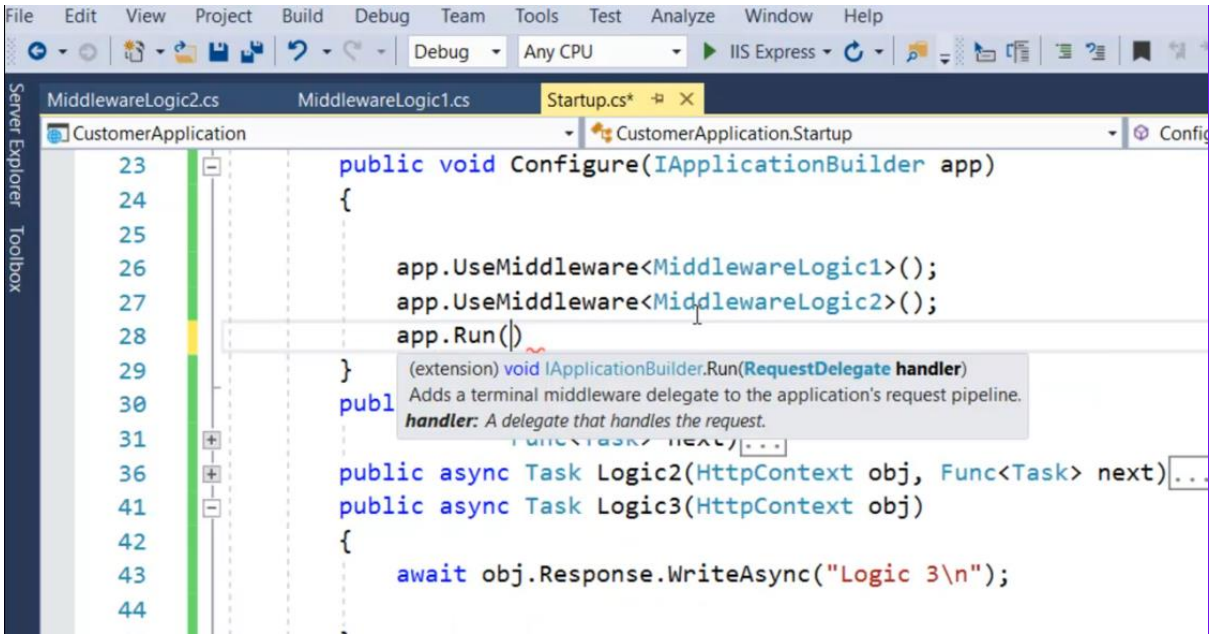


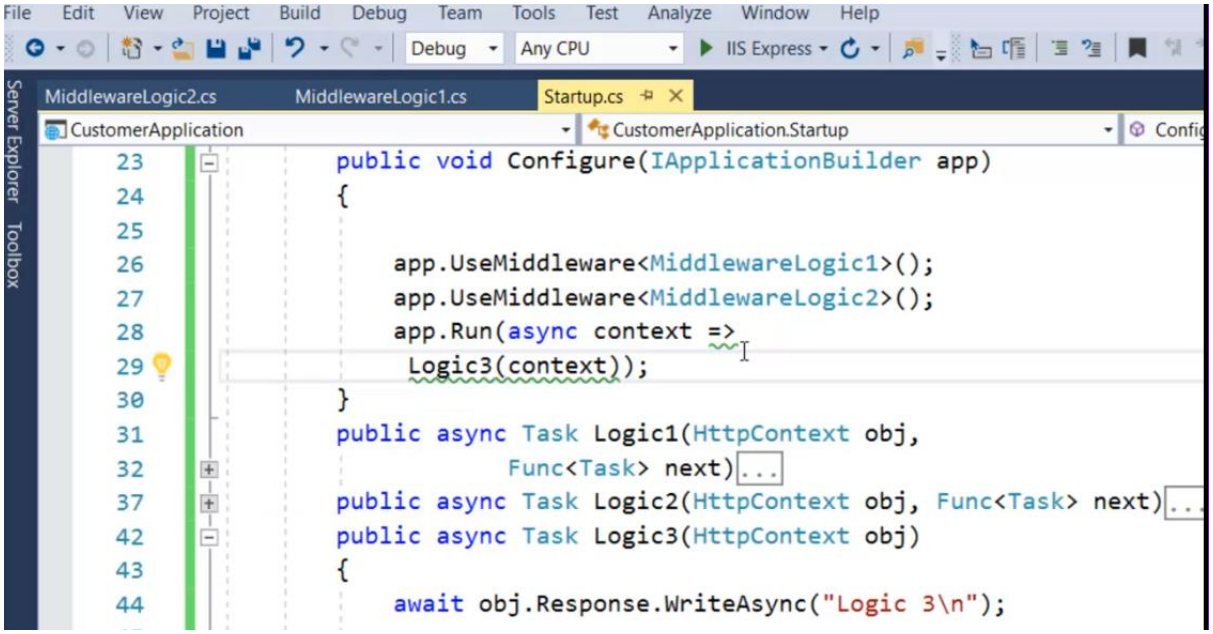
Same with logic2



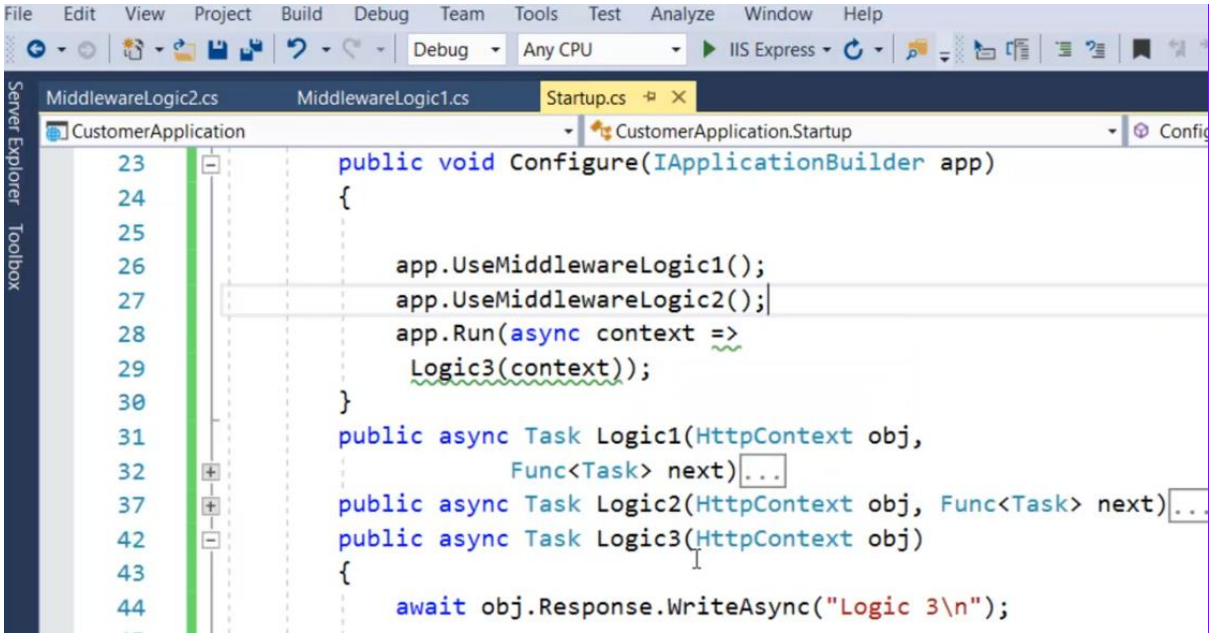
I have to use logic below in place of following

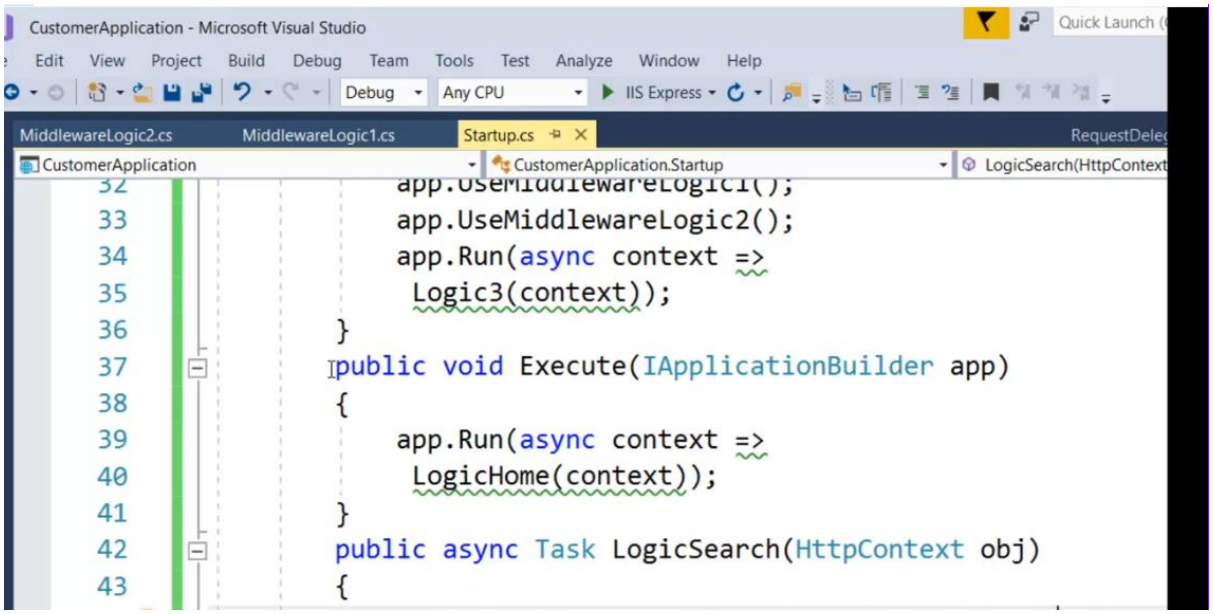

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs*
CustomerApplication
CustomerApplication.Startup
Server Explorer
Toolbox
23 public void Configure(IApplicationBuilder app)
24 {
25
26 app.usem
27 }
28 public
29
34 public
39 public
40 {
41 awa
42
43 }
44 }
UseHttpMethodOverride
UseMicrosoftAccountAuthentication
UseMiddleware
UseMiddleware<>
UseMiddlewareLogic1
UseMiddlewareLogic2
UseMigrationsEndPoint
UseMvc
UseMvcWithDefaultRoute
(extension) IApplicationBuilder IApp
object[] args) (+ 1 generic overload)
Adds a middleware type to the appli
3\n");

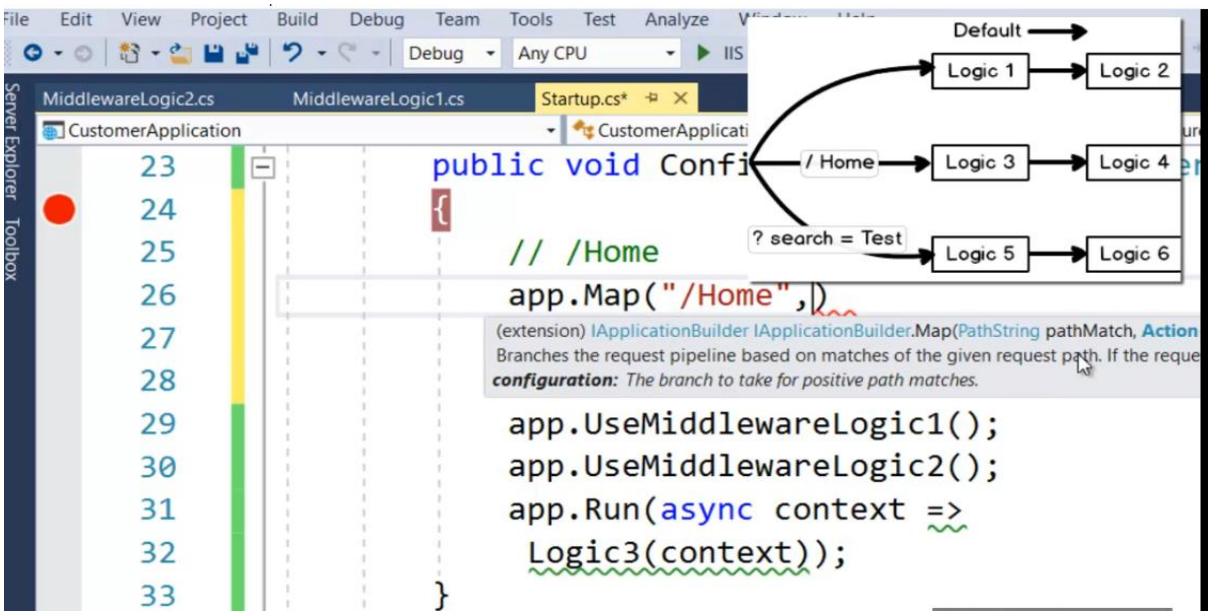

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs*
CustomerApplication
CustomerApplication.Startup
Server Explorer
Toolbox
23 public void Configure(IApplicationBuilder app)
24 {
25
26 app.UseMiddleware<MiddlewareLogic1>();
27 app.UseMiddleware<MiddlewareLogic2>();
28 app.Run()
29 }
30 publ
31
36 public async Task Logic2(HttpContext obj, Func<Task> next)
41 public async Task Logic3(HttpContext obj)
42 {
43 await obj.Response.WriteAsync("Logic 3\n");
44
(extension) void IApplicationBuilder.Run(RequestDelegate handler)
Adds a terminal middleware delegate to the application's request pipeline.
handler: A delegate that handles the request.

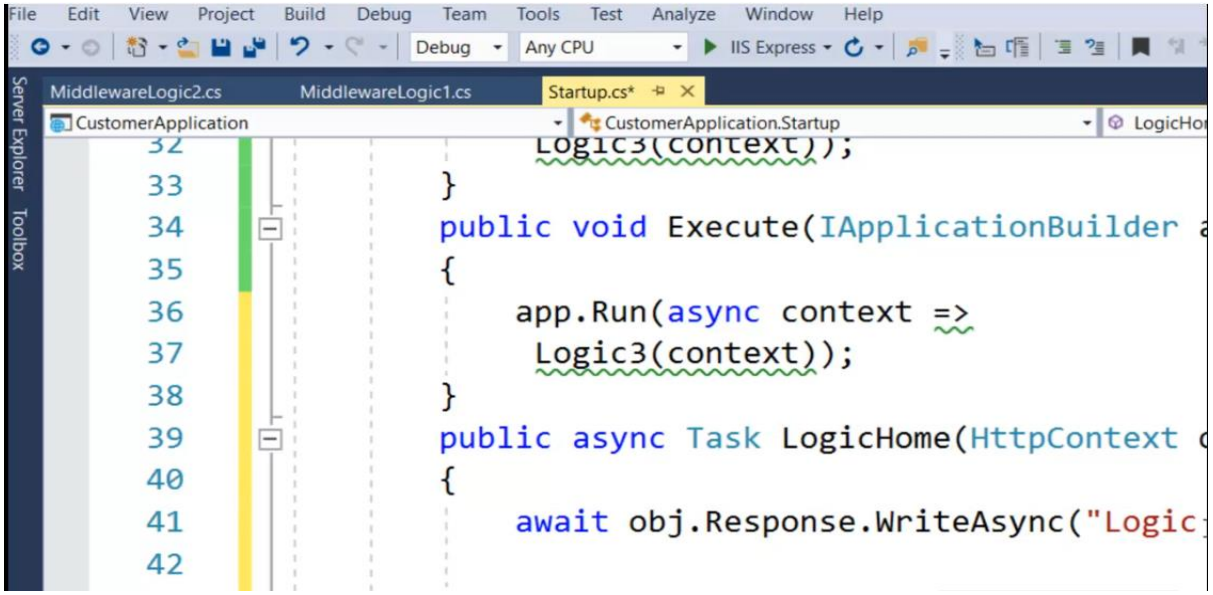


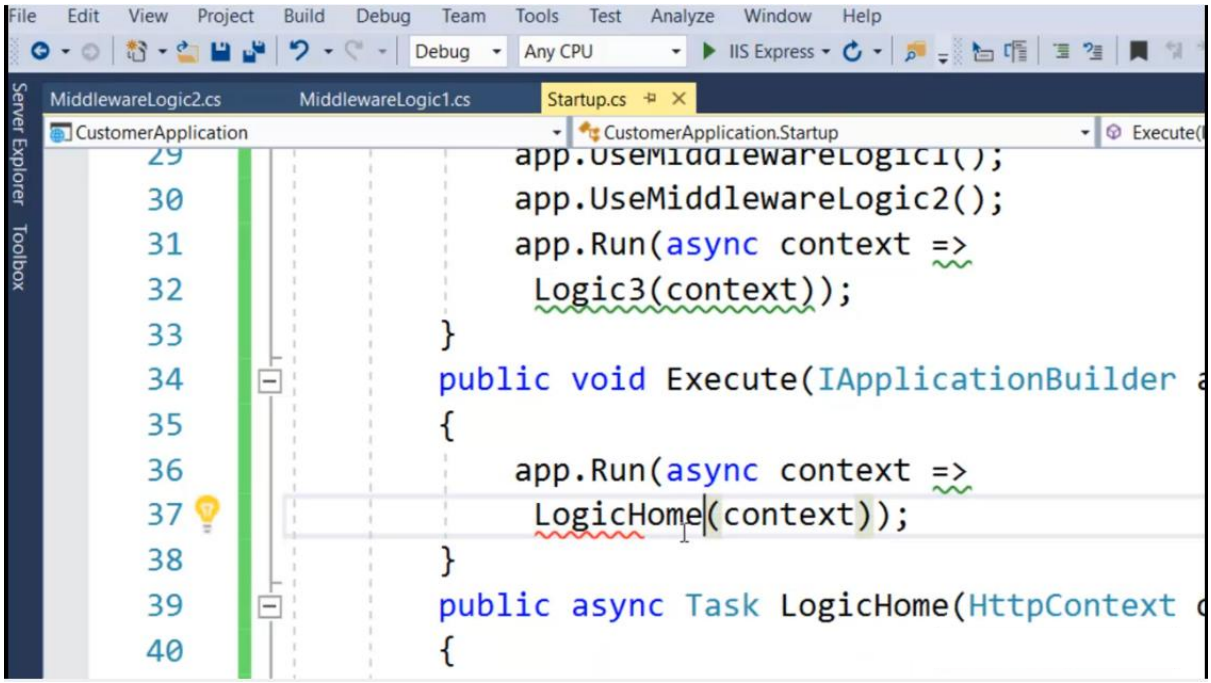
The middle ware is nothing , the logic , that you want put in pipe line.

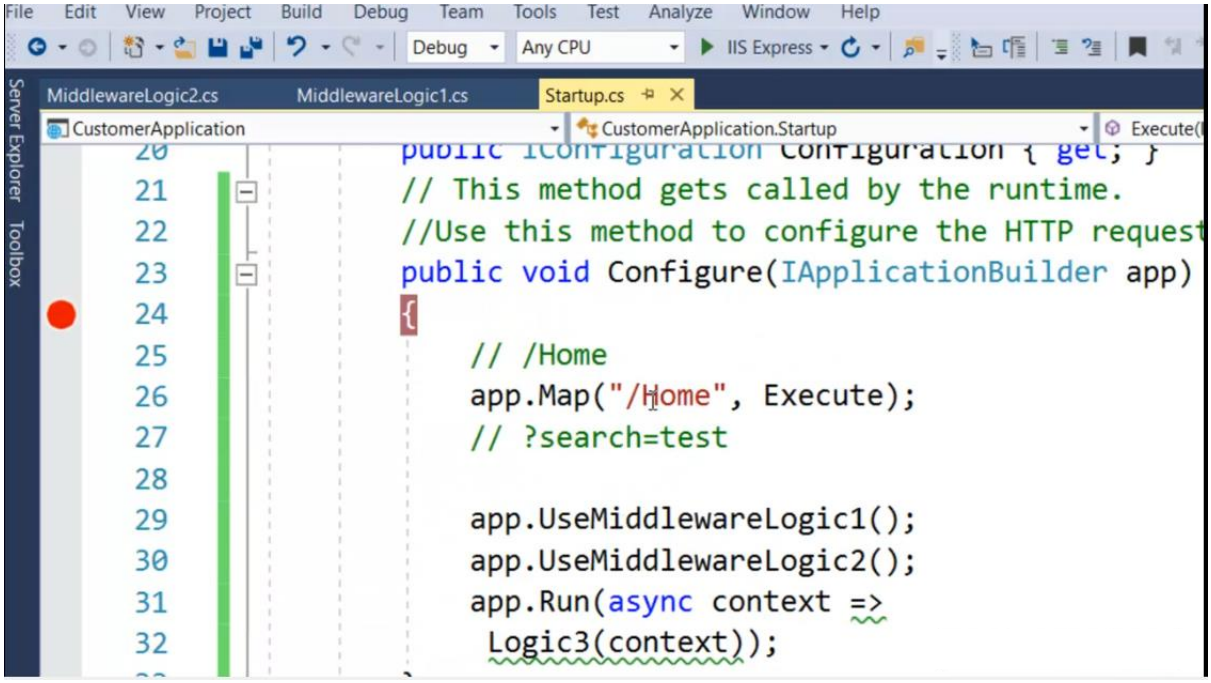


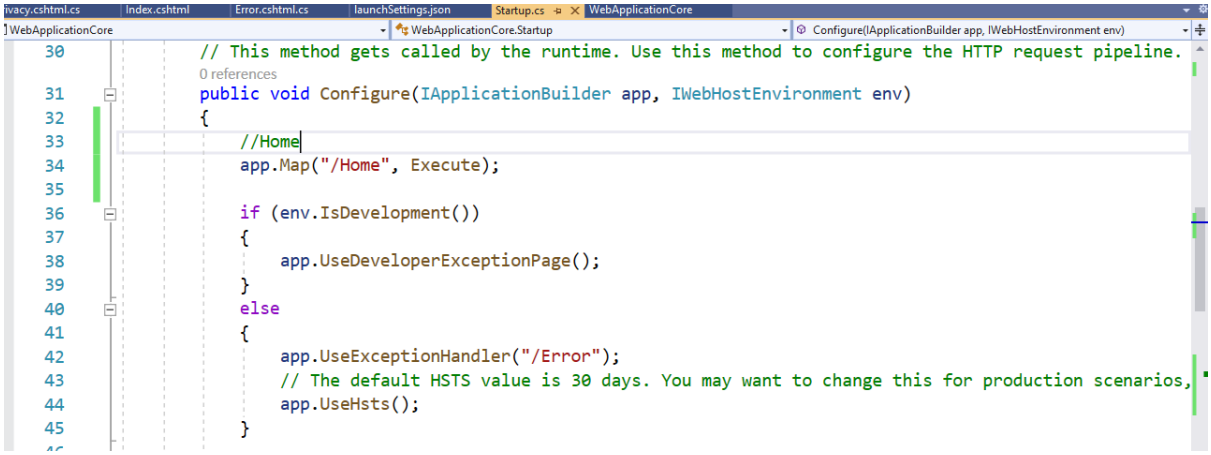

CustomerApplication - Microsoft Visual Studio
Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs
CustomerApplication
CustomerApplication.Startup
LogicSearch(HttpContext
app.UseMiddlewareLogic1();
app.UseMiddlewareLogic2();
app.Run(async context =>
Logic3(context));
}
public void Execute(IApplicationBuilder app)
{
app.Run(async context =>
LogicHome(context));
}
public async Task LogicSearch(HttpContext obj)
{


Default
Logic 1
Logic 2
/ Home
Logic 3
Logic 4
? search = Test
Logic 5
Logic 6
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs*
CustomerApplication
public void Confi
{
// /Home
app.Map("/Home",
(extension) IApplicationBuilder IApplicationBuilder.Map(PathString pathMatch, Action
Branches the request pipeline based on matches of the given request path. If the reque
configuration: The branch to take for positive path matches.
app.UseMiddlewareLogic1();
app.UseMiddlewareLogic2();
app.Run(async context =>
Logic3(context));
}

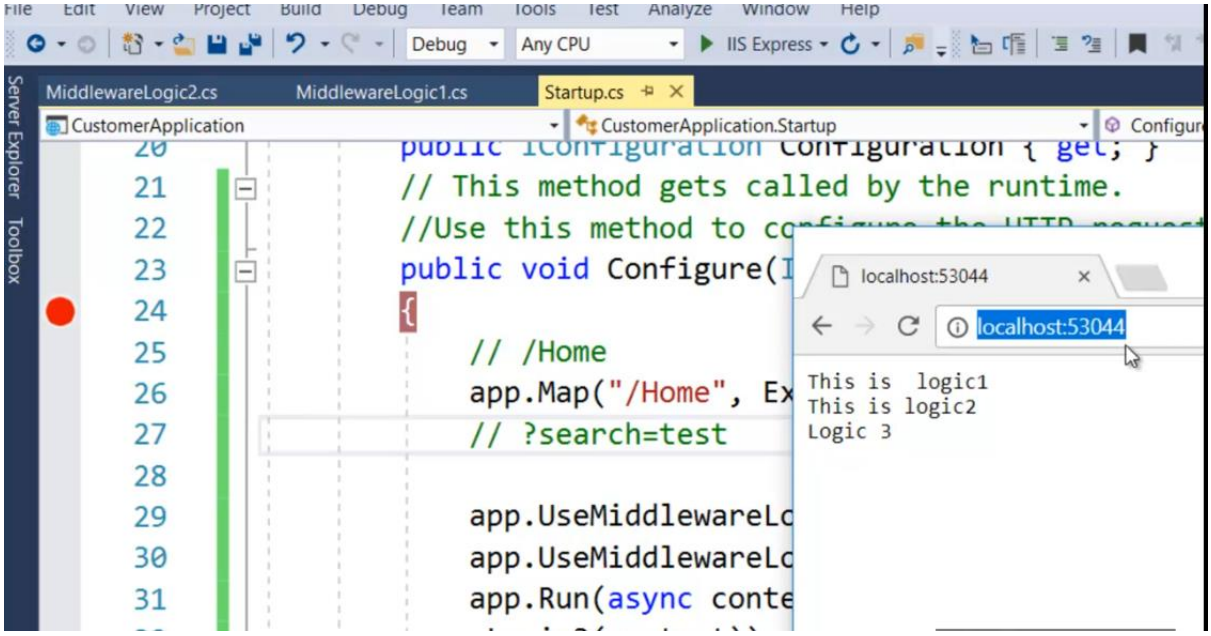

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs*
CustomerApplication
CustomerApplication.Startup
Logic3(context));
}
public void Execute(IApplicationBuilder
{
app.Run(async context =>
Logic3(context));
}
public async Task LogicHome(HttpContext
{
await obj.Response.WriteAsync("Logic

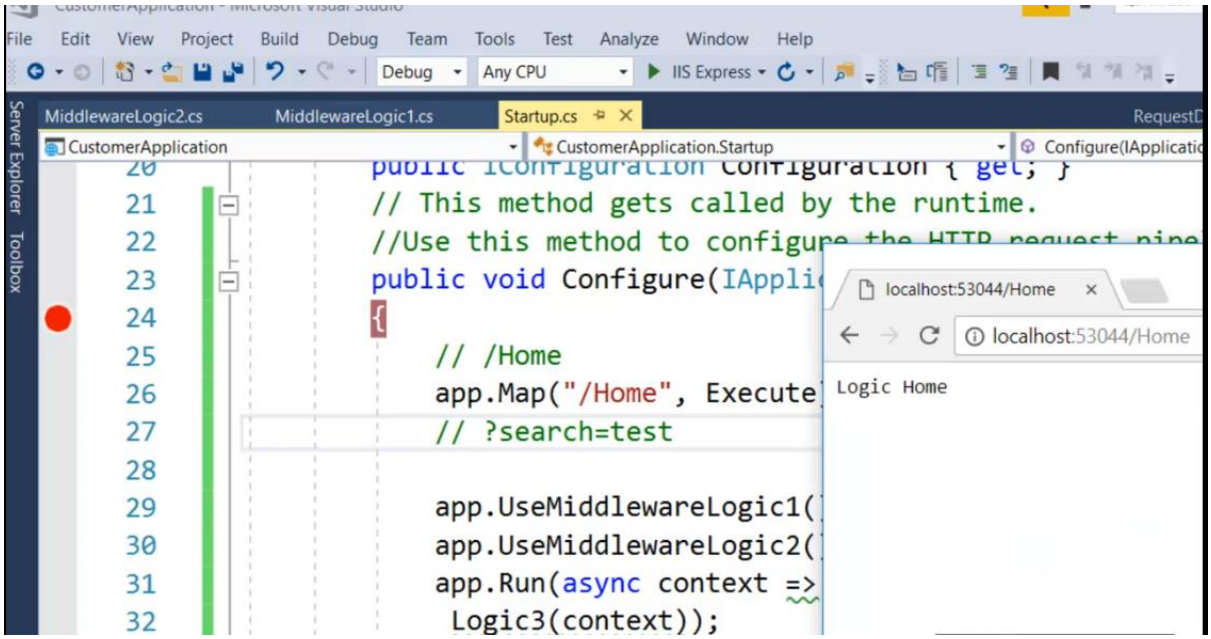

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs
CustomerApplication
CustomerApplication.Startup
Server Explorer
Toolbox
app.UseMiddlewareLogic1();
app.UseMiddlewareLogic2();
app.Run(async context =>
Logic3(context));
}
public void Execute(IApplicationBuilder
{
app.Run(async context =>
LogicHome(context));
}
public async Task LogicHome(HttpContext
{

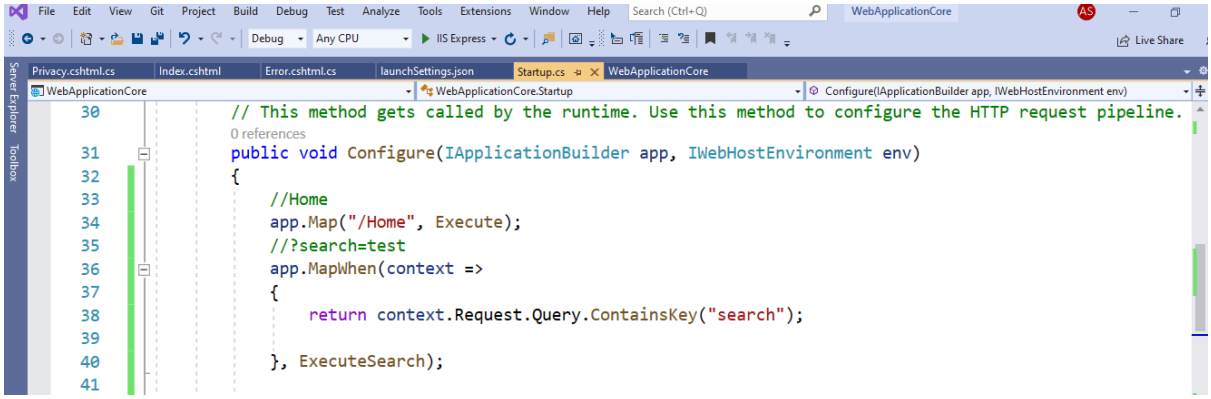

File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
MiddlewareLogic2.cs
MiddlewareLogic1.cs
Startup.cs
CustomerApplication
CustomerApplication.Startup
Server Explorer
Toolbox
// This method gets called by the runtime.
//Use this method to configure the HTTP request
public void Configure(IApplicationBuilder app)
{
// /Home
app.Map("/Home", Execute);
// ?search=test
app.UseMiddlewareLogic1();
app.UseMiddlewareLogic2();
app.Run(async context =>
Logic3(context));


Startup.cs
WebApplicationCore
WebApplicationCore.Startup
Configure(IApplicationBuilder app, IWebHostEnvironment env)
// This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
0 references
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
//Home
app.Map("/Home", Execute);
if (env.IsDevelopment())
{
app.UseDeveloperExceptionPage();
}
else
{
app.UseExceptionHandler("/Error");
// The default HSTS value is 30 days. You may want to change this for production scenarios,
app.UseHsts();
}

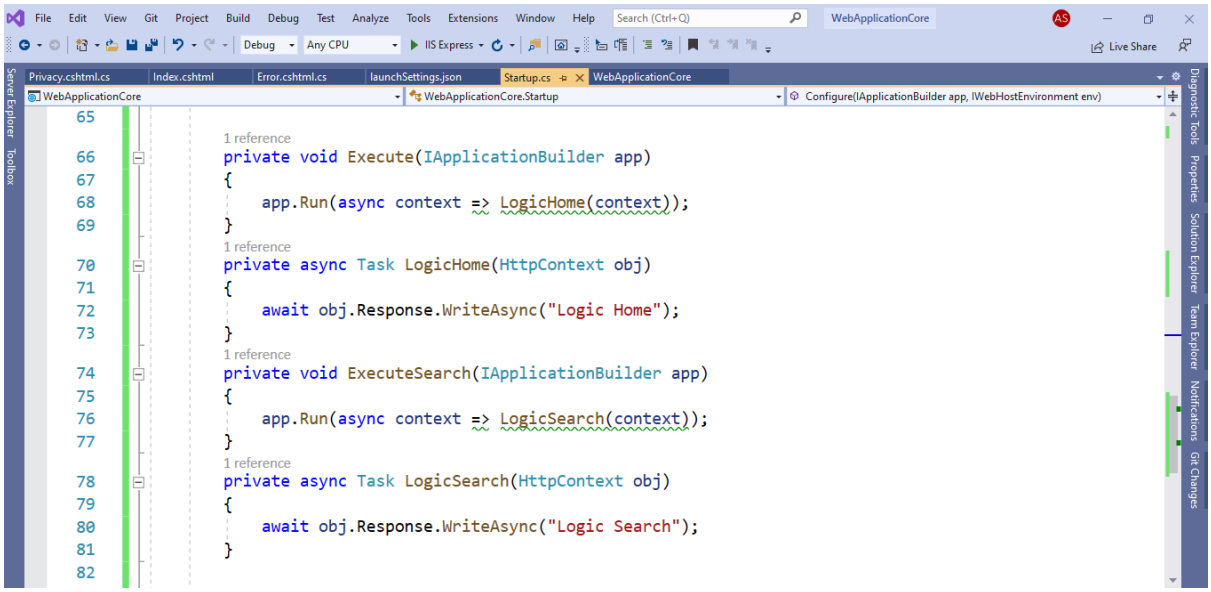
So, if you want to build your pipe line, using the URL, using the query string, then you need to use map or map when.



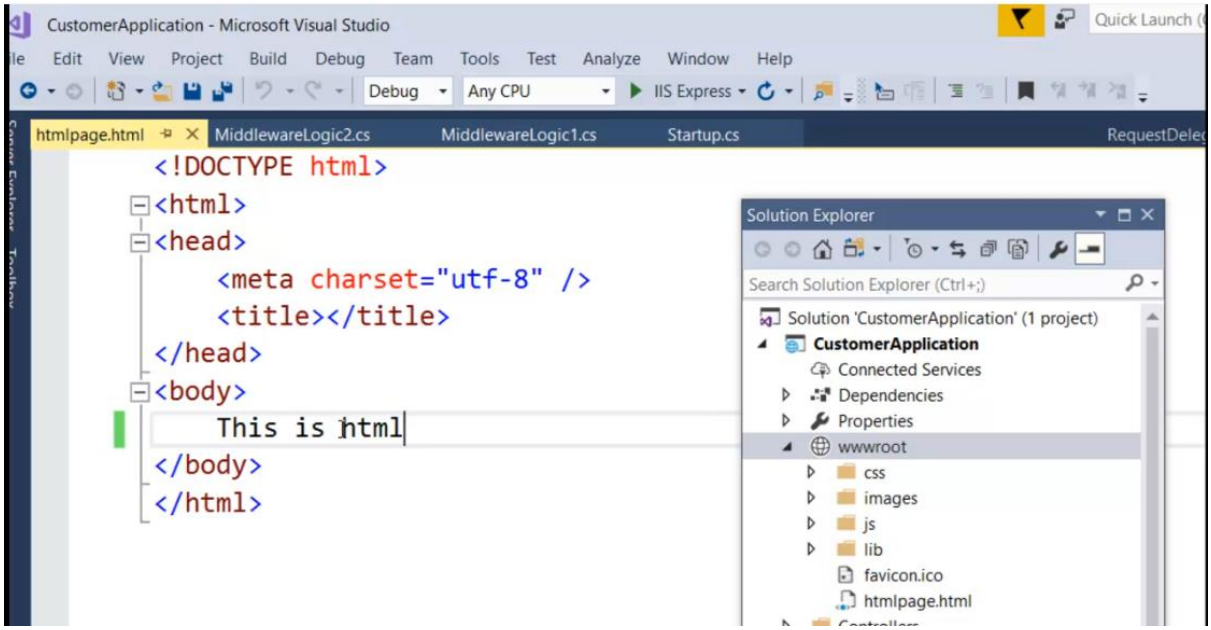




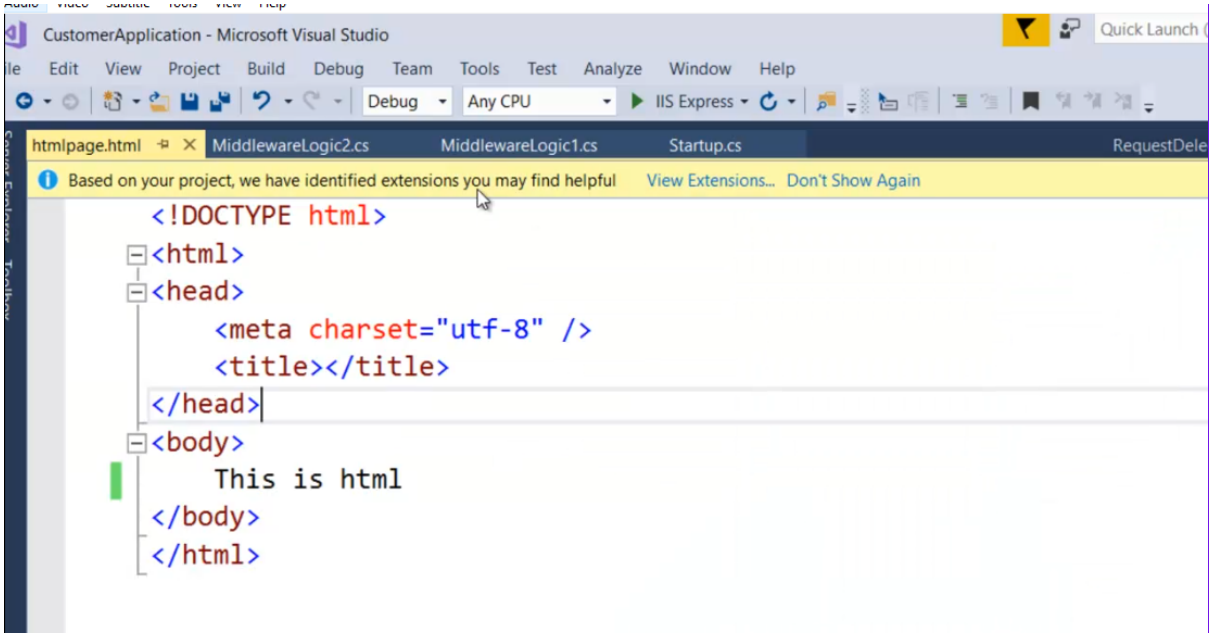
```
30          // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
            0 references
31          public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
32          {
33              //Home
34              app.Map("/Home", Execute);
35              //?search=test
36              app.MapWhen(context =>
37              {
38                  return context.Request.Query.ContainsKey("search");
39
40              }, ExecuteSearch);
41
```

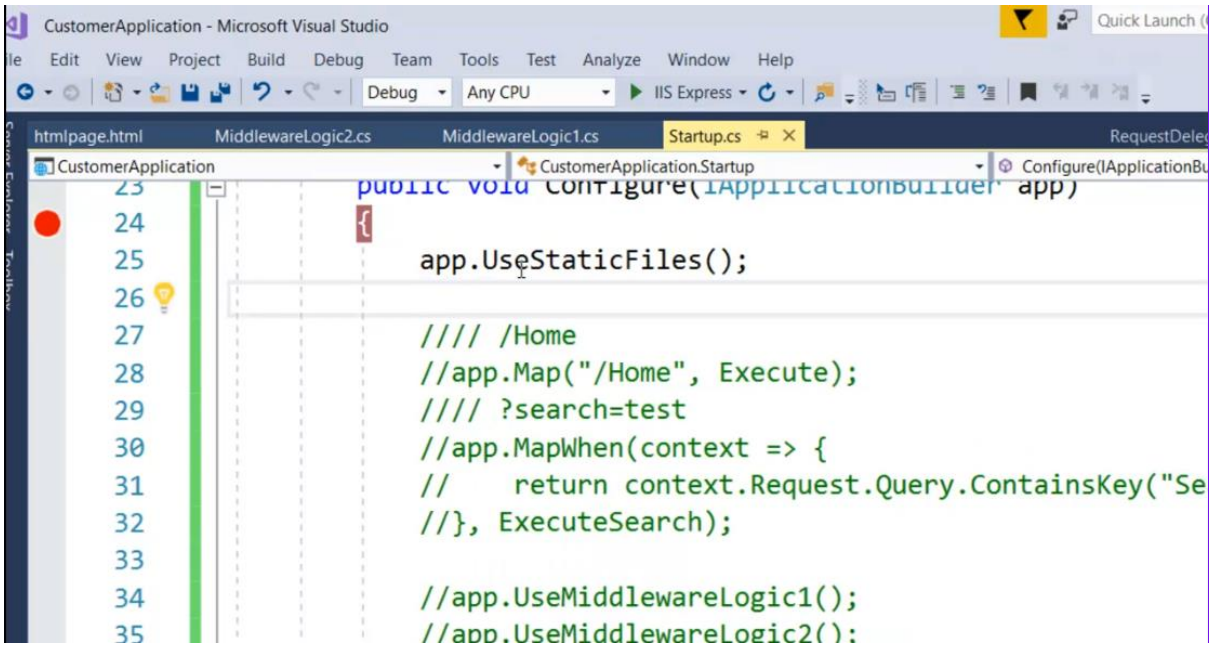


```
65
            1 reference
66          private void Execute(IApplicationBuilder app)
67          {
68              app.Run(async context => LogicHome(context));
69          }
            1 reference
70          private async Task LogicHome(HttpContext obj)
71          {
72              await obj.Response.WriteAsync("Logic Home");
73          }
            1 reference
74          private void ExecuteSearch(IApplicationBuilder app)
75          {
76              app.Run(async context => LogicSearch(context));
77          }
            1 reference
78          private async Task LogicSearch(HttpContext obj)
79          {
80              await obj.Response.WriteAsync("Logic Search");
81          }
82
```

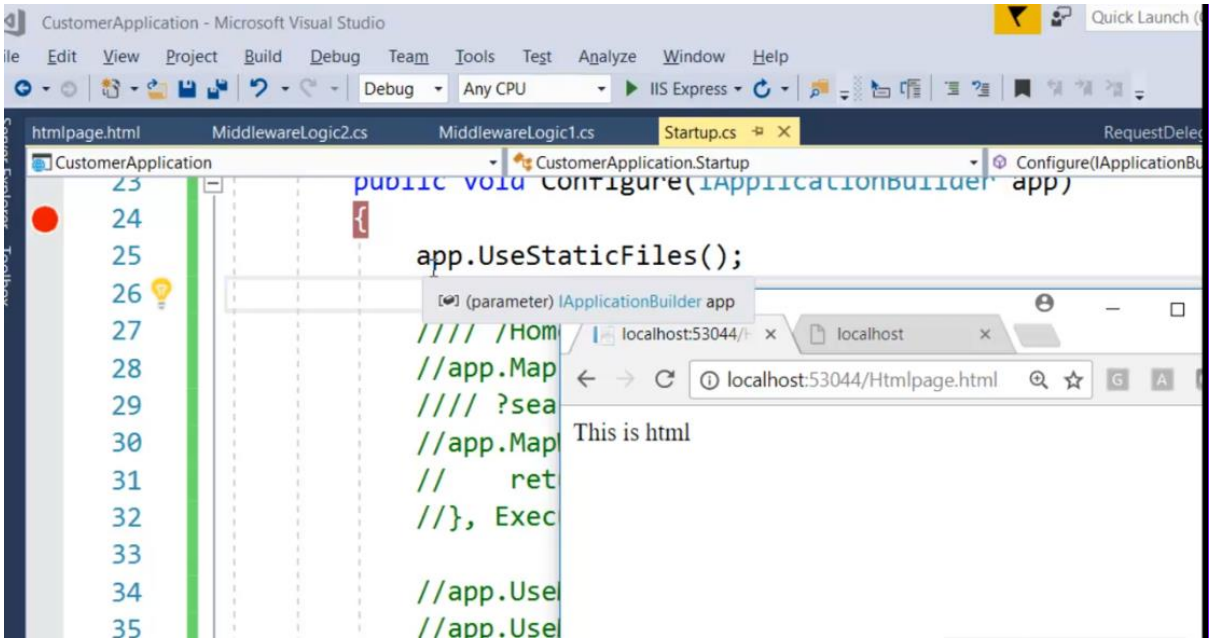


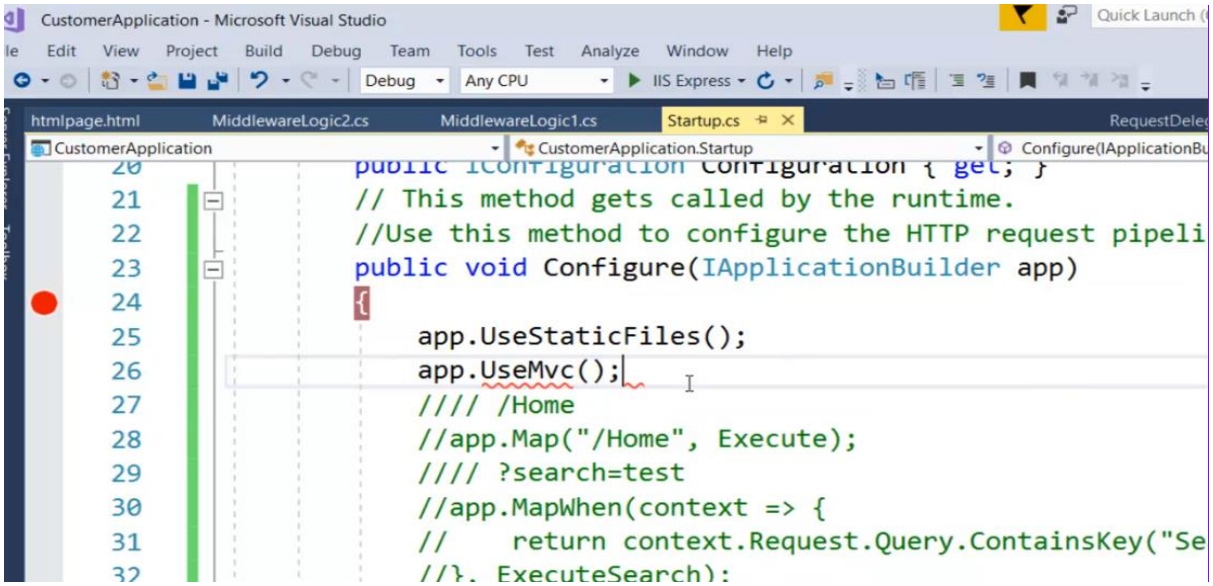
```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title></title>
</head>
<body>
    This is html
</body>
</html>
```





Note : Static file should be wwwroot folder



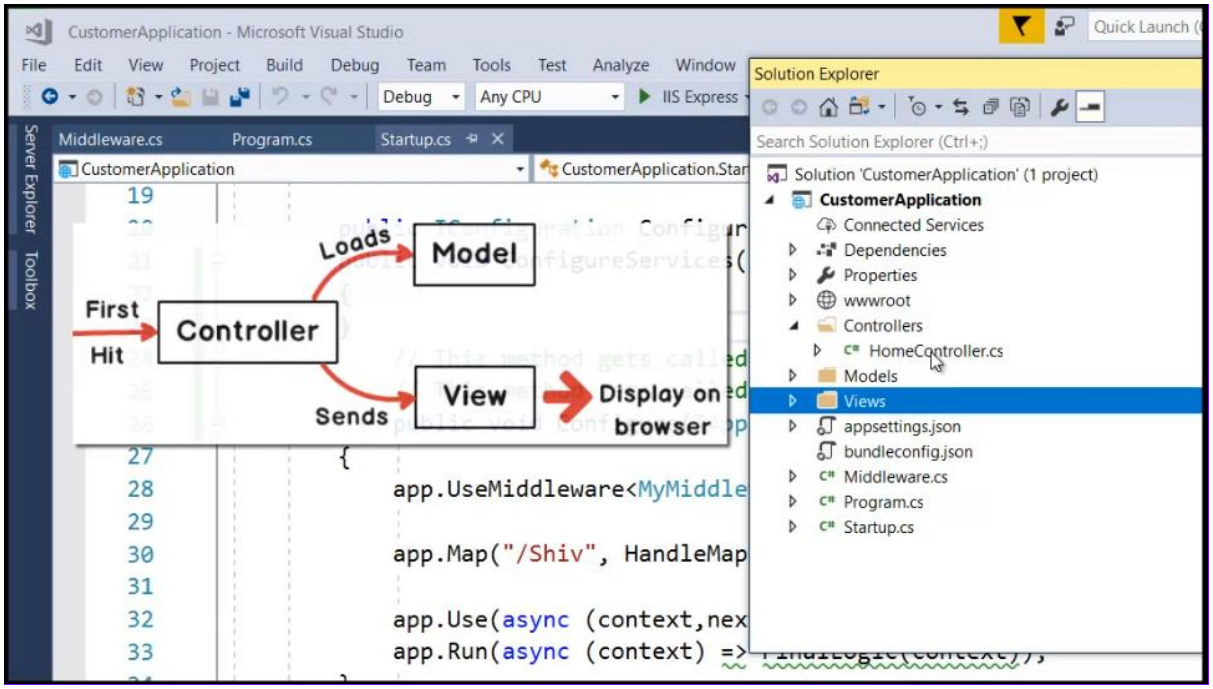


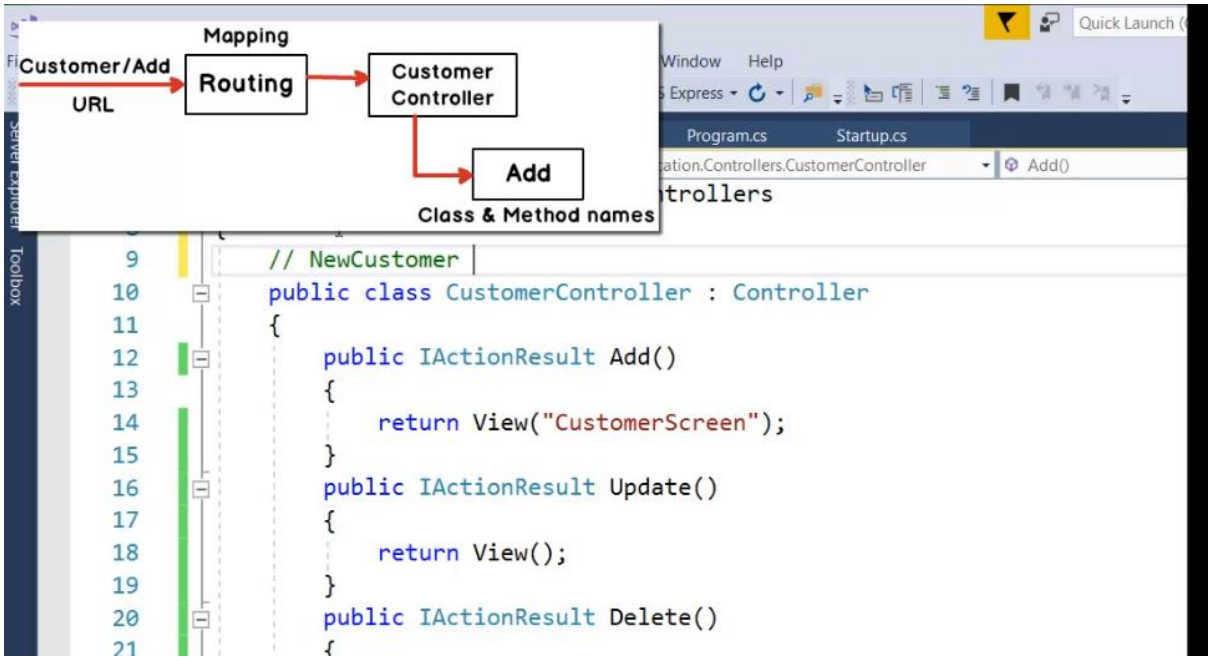
Lets test what you have learnt.

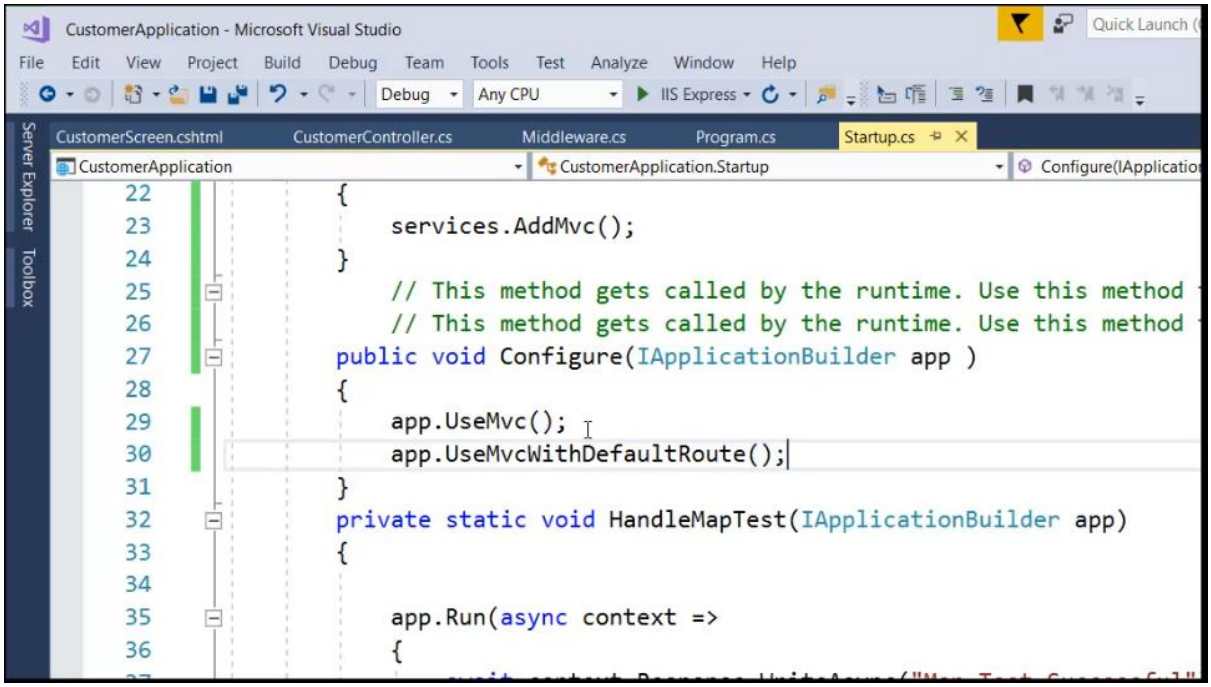
Define context , request and response ?
Explain terms request pipeline and middleware?
In MVC core where do we configure request pipeline?
What is the importance of "IApplicationBuilder"?
Explain "use" , "map" and "run" ?
How can you create middleware using classes?
Differentiate between "map" and "mapwhen" ?

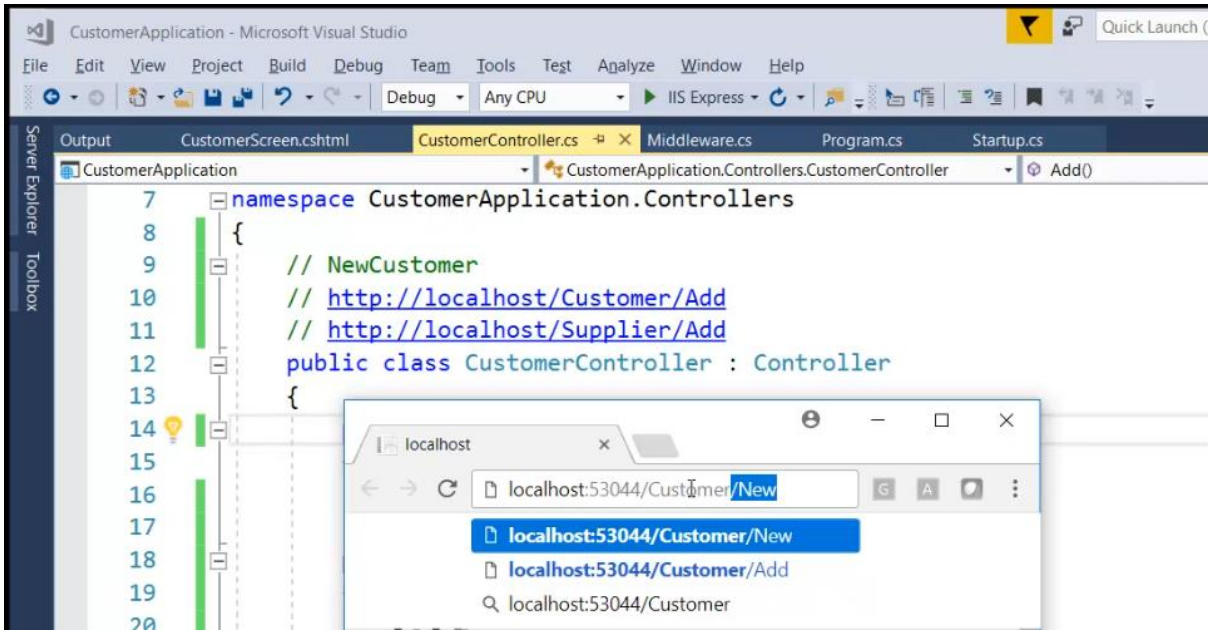
Implementing basic Model, View & Controller

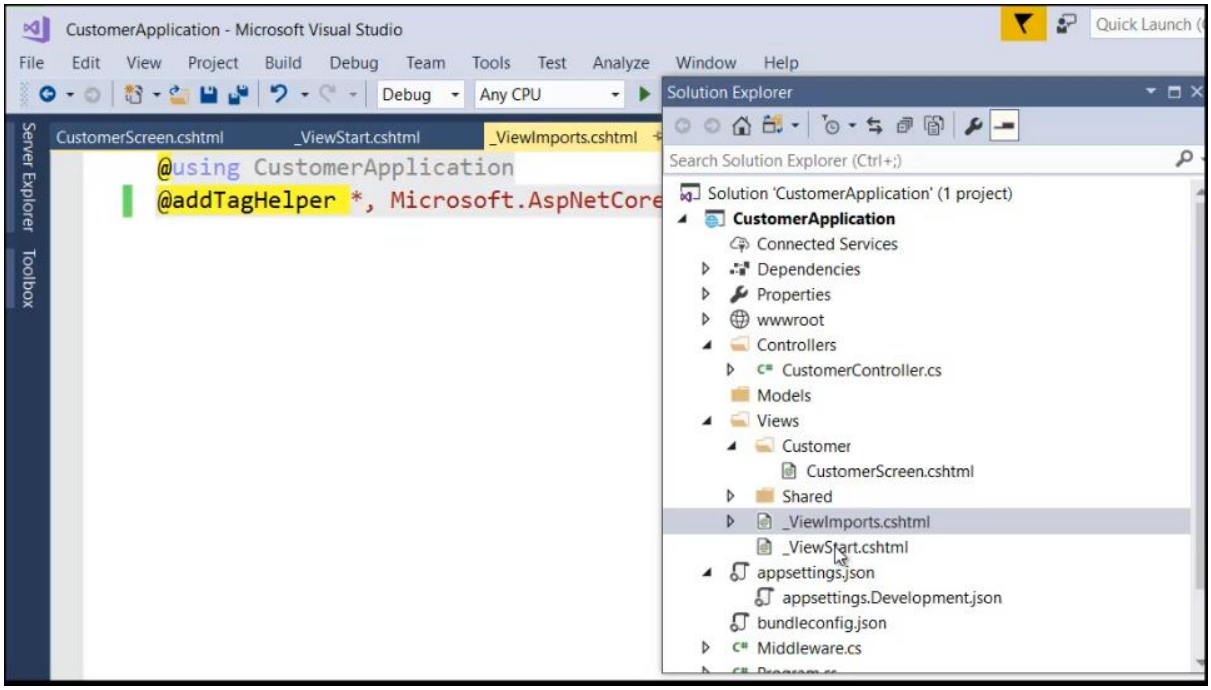
Create Controllers and Views
Understanding ActionResults in MVC
What is Scaffolding ?
MVC Core namespaces
Layout , Routing and Razor pages

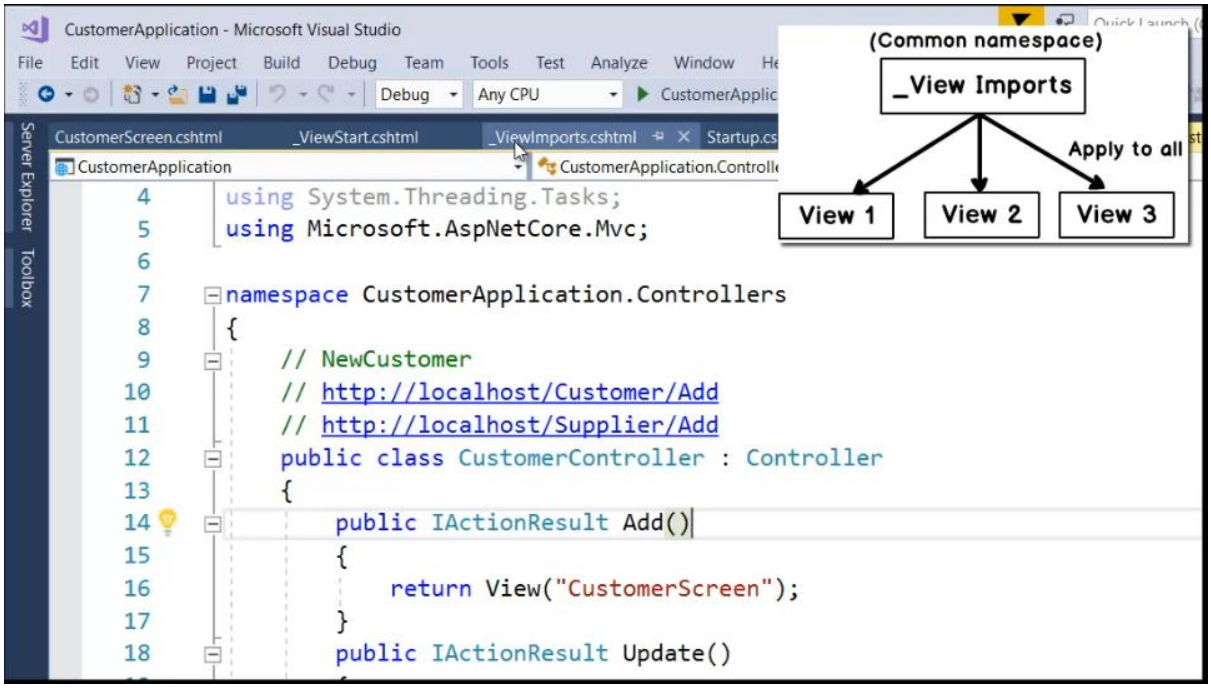



Mapping
Customer/Add
URL
Routing
Customer Controller
Add
Class & Method names
Program.cs
Startup.cs
Add()
// NewCustomer
public class CustomerController : Controller
{
public IActionResult Add()
{
return View("CustomerScreen");
}
public IActionResult Update()
{
return View();
}
public IActionResult Delete()

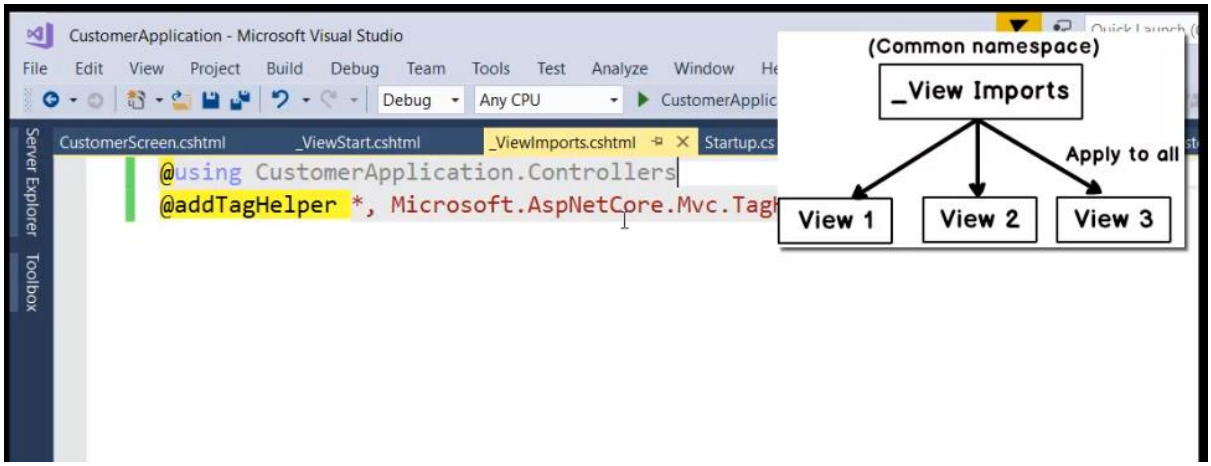

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
CustomerScreen.cshtml
CustomerController.cs
Middleware.cs
Program.cs
Startup.cs
CustomerApplication
CustomerApplication.Startup
services.AddMvc();
// This method gets called by the runtime. Use this method
// This method gets called by the runtime. Use this method
public void Configure(IApplicationBuilder app)
app.UseMvc();
app.UseMvcWithDefaultRoute();
private static void HandleMapTest(IApplicationBuilder app)
app.Run(async context =>

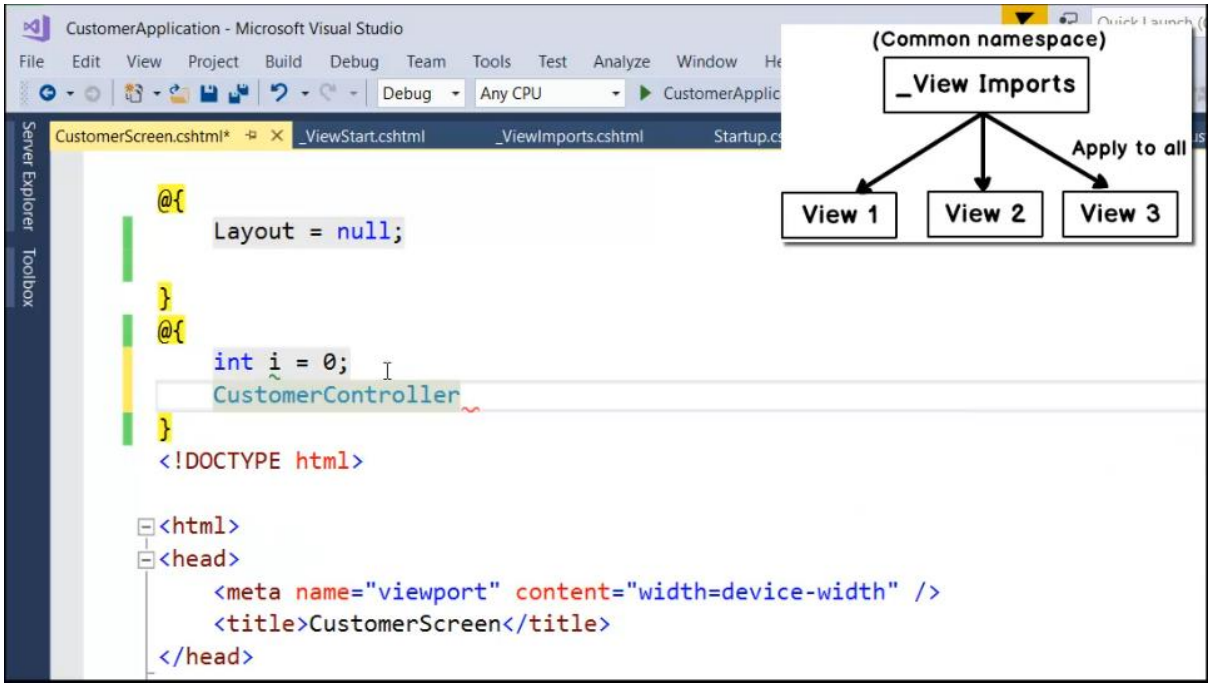

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Output
CustomerScreen.cshtml
CustomerController.cs
Middleware.cs
Program.cs
Startup.cs
CustomerApplication
CustomerApplication.Controllers.CustomerController
Add()
Server Explorer
Toolbox
namespace CustomerApplication.Controllers
{
// NewCustomer
// http://localhost/Customer/Add
// http://localhost/Supplier/Add
public class CustomerController : Controller
{
localhost
localhost:53044/Customer/New
localhost:53044/Customer/New
localhost:53044/Customer/Add
localhost:53044/Customer

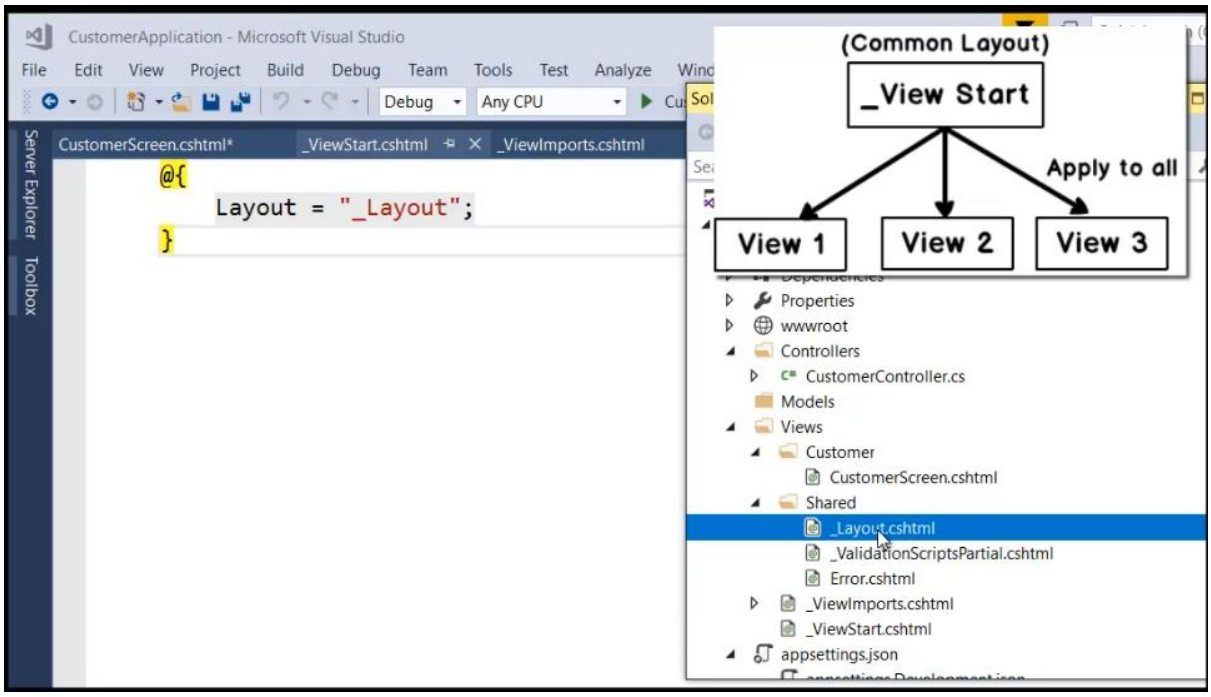

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Solution Explorer
Search Solution Explorer (Ctrl+;)
CustomerScreen.cshtml
_ViewStart.cshtml
_ViewImports.cshtml
@using CustomerApplication
@addTagHelper *, Microsoft.AspNetCore
Solution 'CustomerApplication' (1 project)
CustomerApplication
Connected Services
Dependencies
Properties
wwwroot
Controllers
CustomerController.cs
Models
Views
Customer
CustomerScreen.cshtml
Shared
_ViewImports.cshtml
_ViewStart.cshtml
appsettings.json
appsettings.Development.json
bundleconfig.json
Middleware.cs



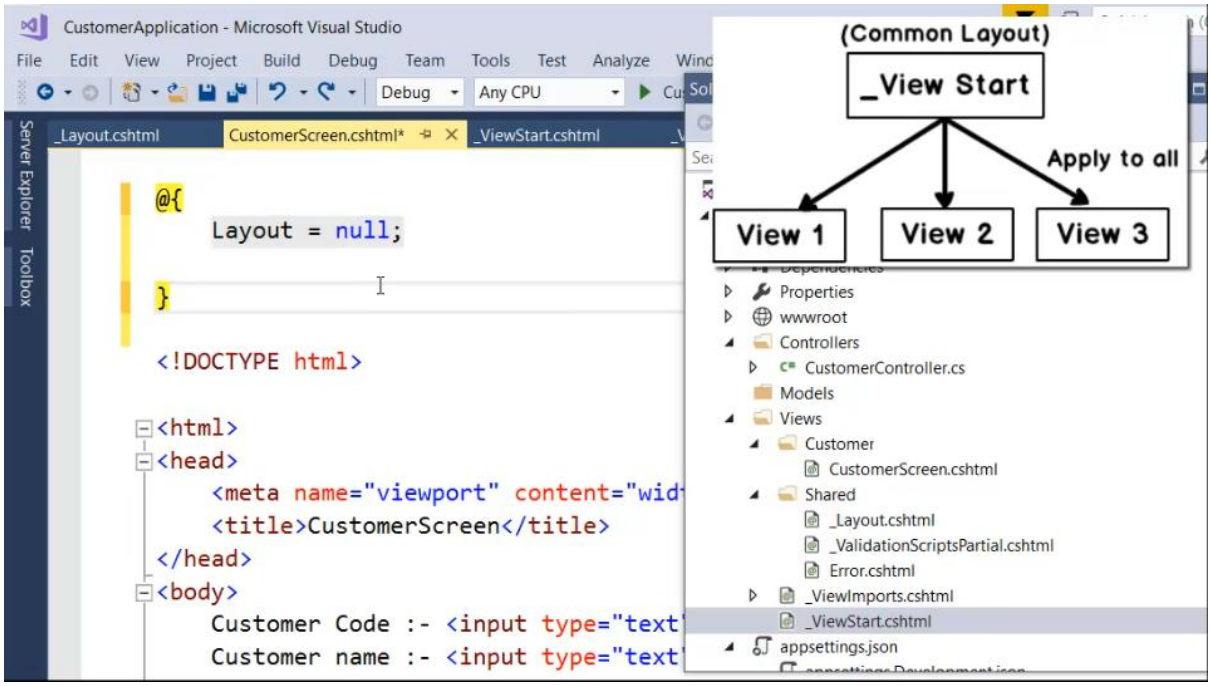
Note : If you add any namespace In the view imports , then it will be access by across view.







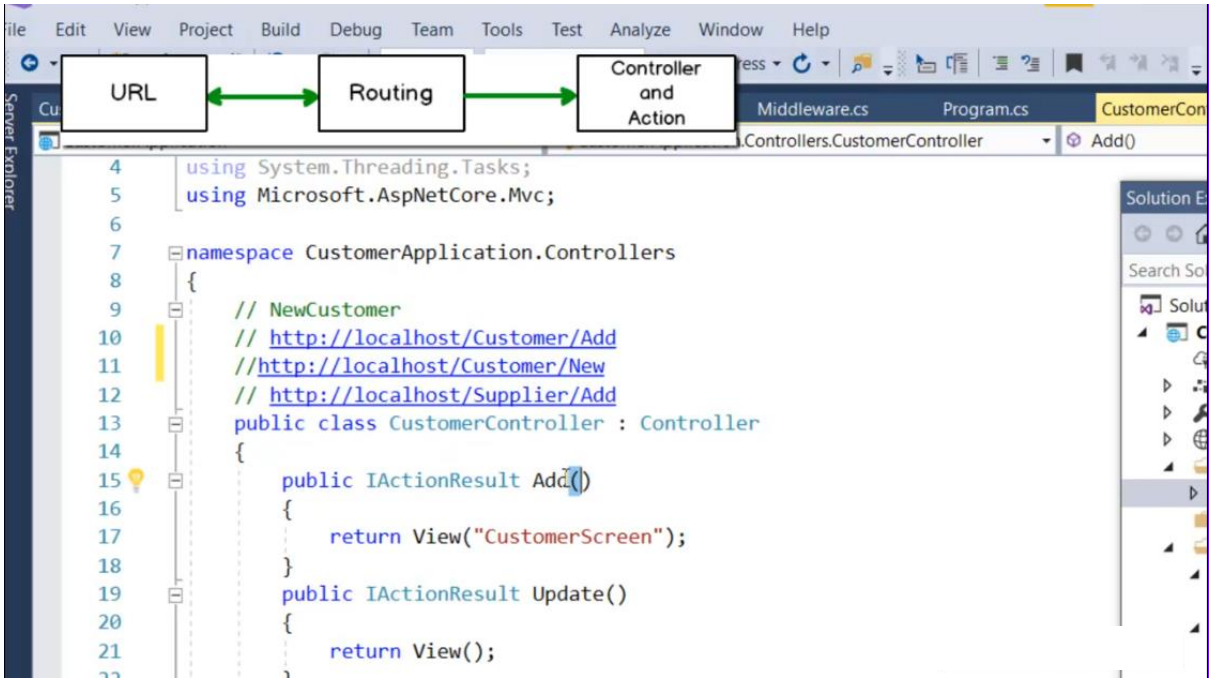
If no layout will be apply on , so by default layout will apply.

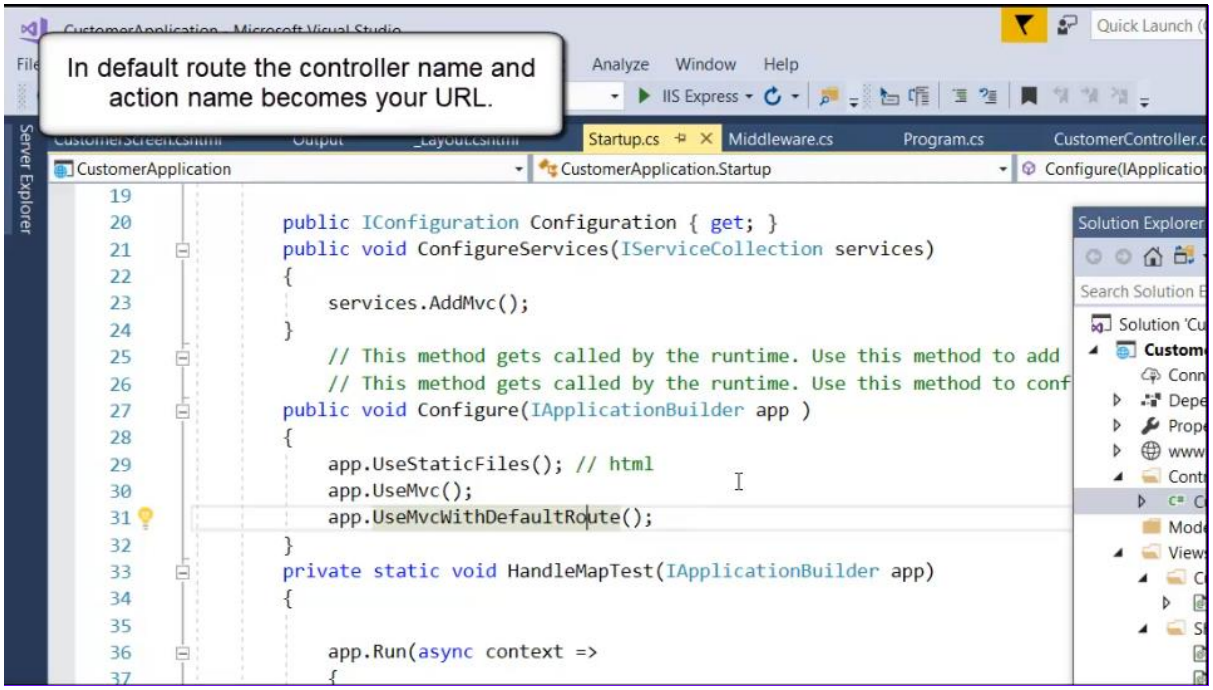


Lets revise what we have learnt.....

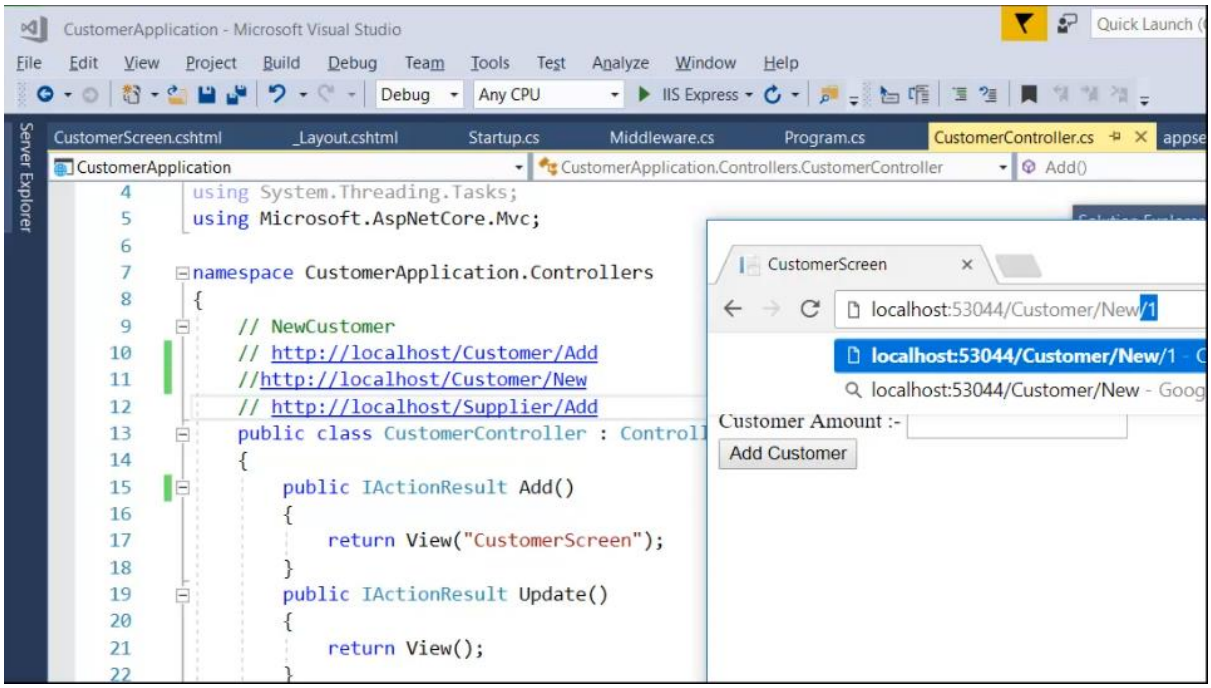
Which are the reserved folders for controllers , views and models ?
What is the naming conventions of MVC controller classes ?
What steps do we need to add MVC services to MVC core project ?
Explain the difference between "Services.AddMvc" and "app.UseMVC" ?
What are actions in Controller ?
What is "IActionResult" in MVC Controller ?
Explain the term scaffolding ?
What are the two different types of controllers in MVC ?
All MVC controller class inherit from _____ class.
Which is the main namespace of MVC core ?
Explain layout pages ?
What is routing in MVC and how to enable default routes ?
Explain Razor page (CSHTML) ?

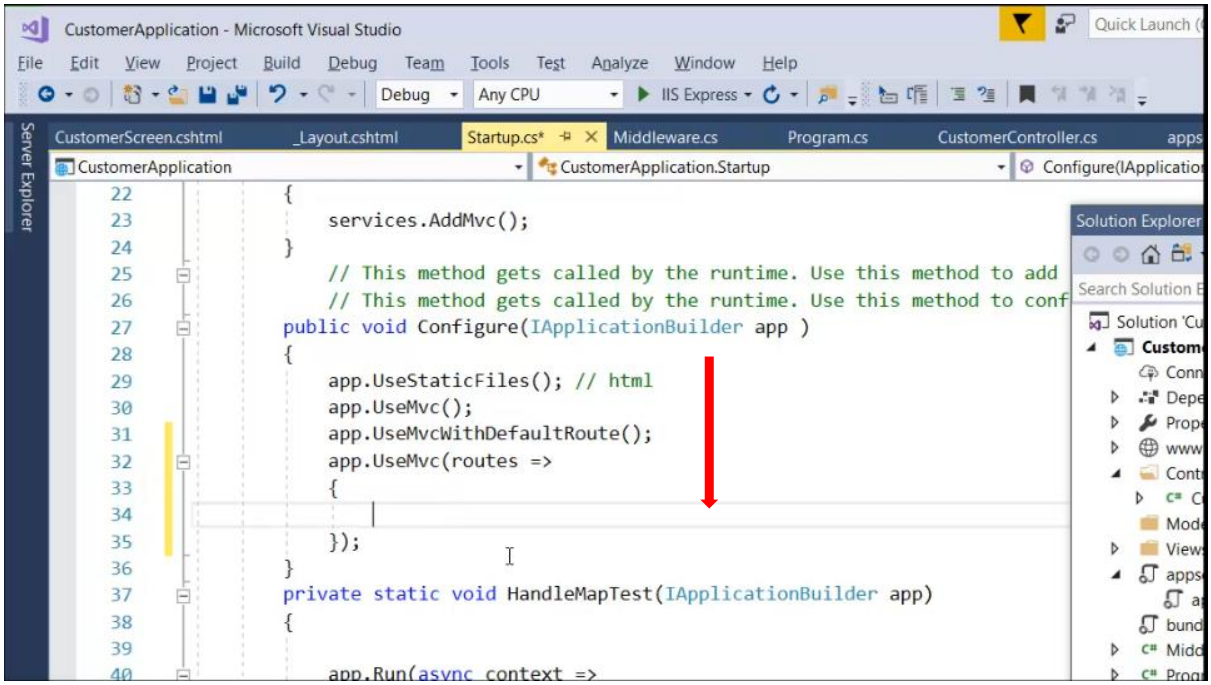
Routing,Conventional,Attribute and Route constraint

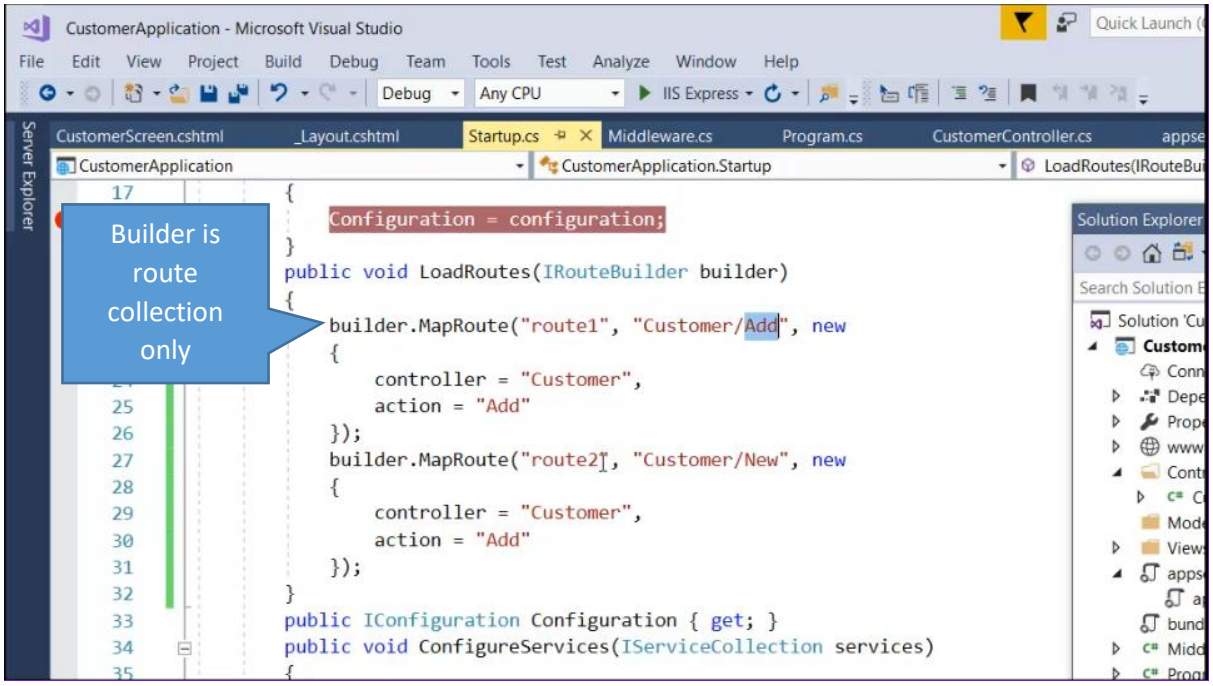


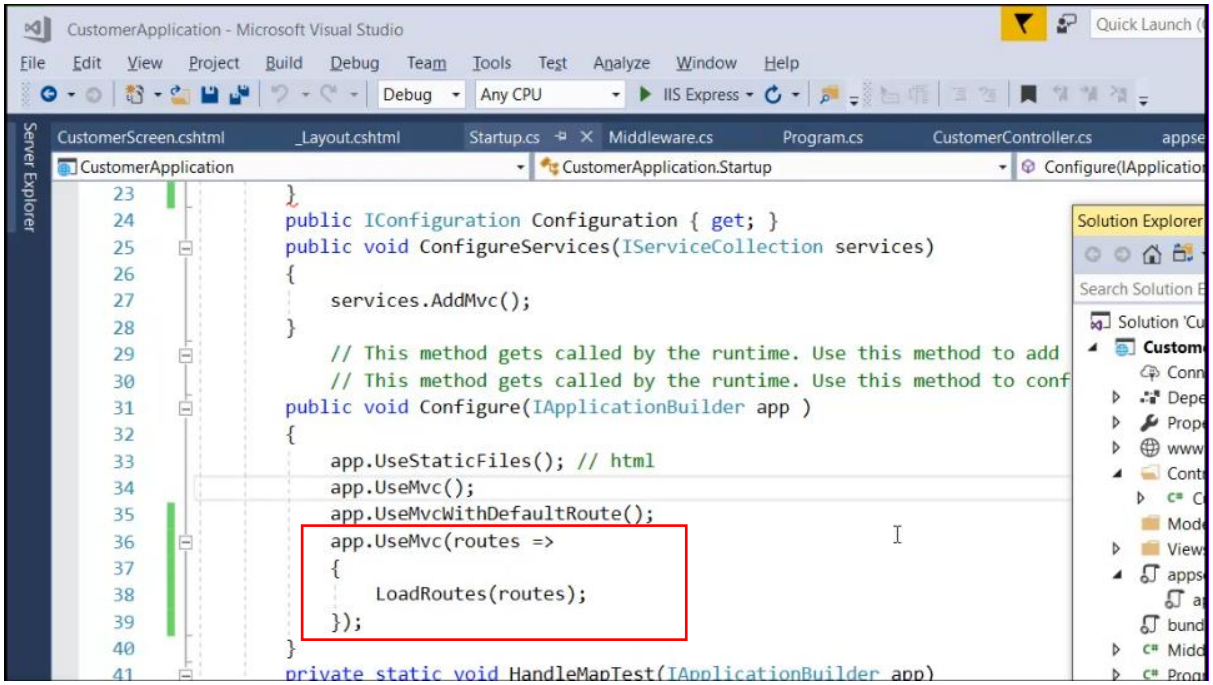


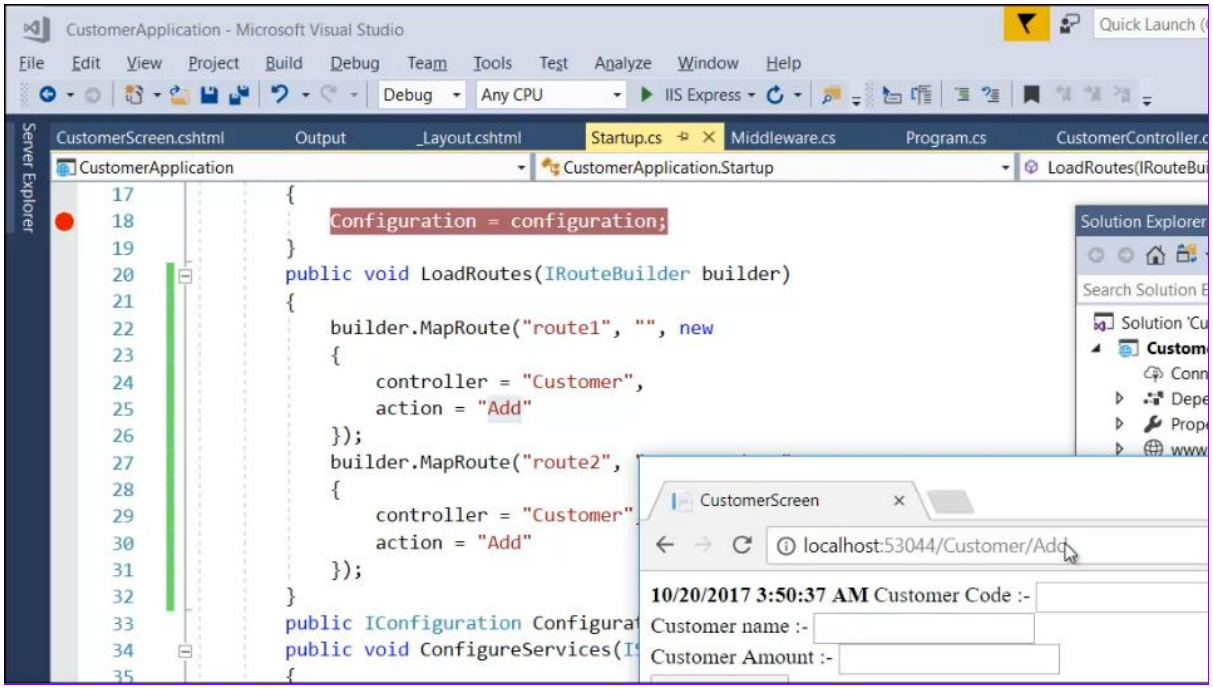
Now if we are looking for route like



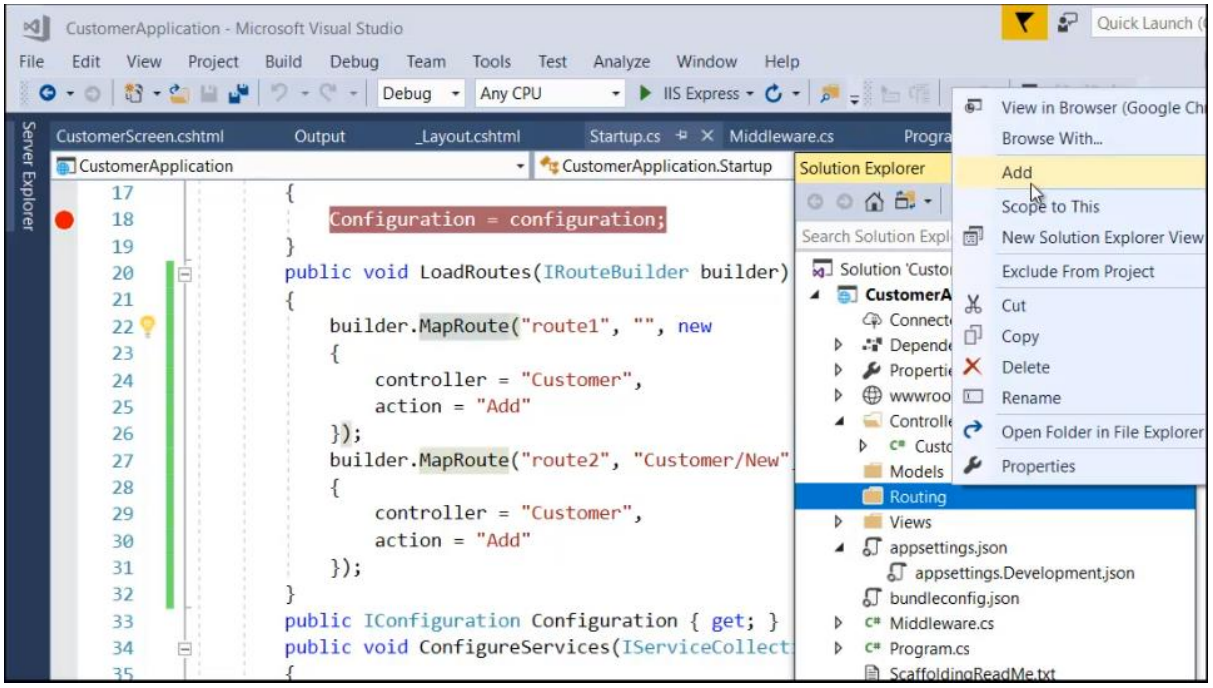


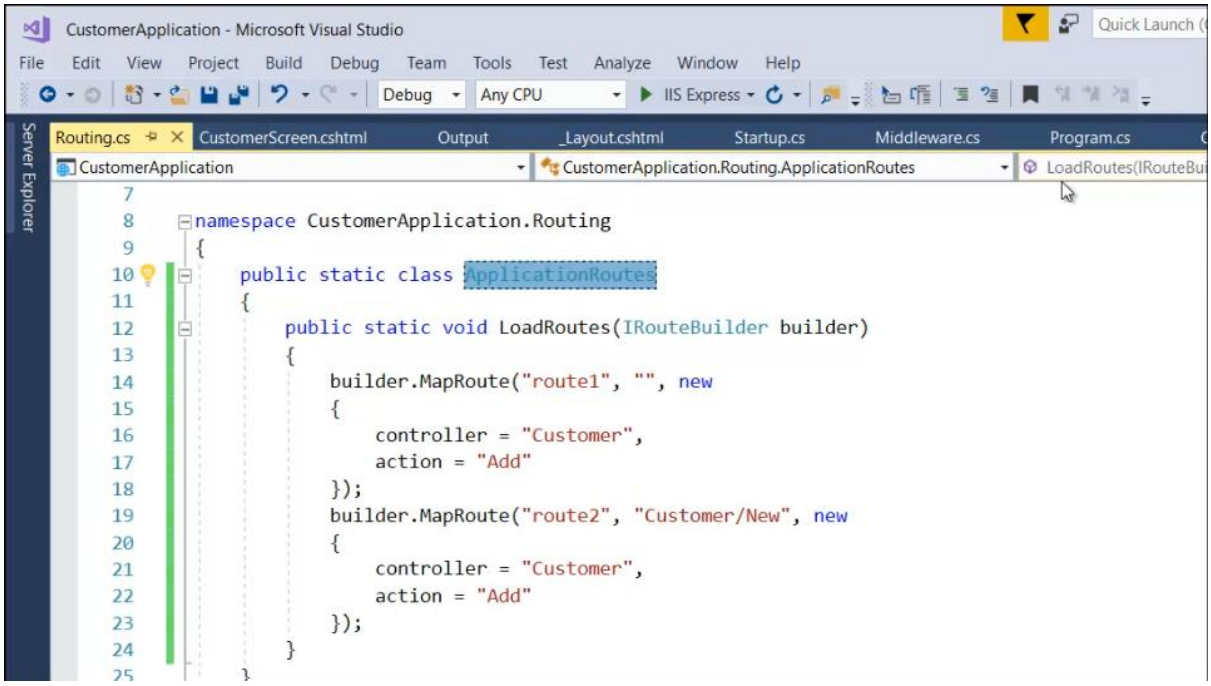

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
CustomerScreen.cshtml
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
CustomerController.cs
CustomerApplication
CustomerApplication.Startup
LoadRoutes(IRouteBui
Builder is route collection only
Configuration = configuration;
}
public void LoadRoutes(IRouteBuilder builder)
{
builder.MapRoute("route1", "Customer/Add", new
{
controller = "Customer",
action = "Add"
});
builder.MapRoute("route2", "Customer/New", new
{
controller = "Customer",
action = "Add"
});
}
public IConfiguration Configuration { get; }
public void ConfigureServices(IServiceCollection services)
{
Solution Explorer
Search Solution E
Server Explorer

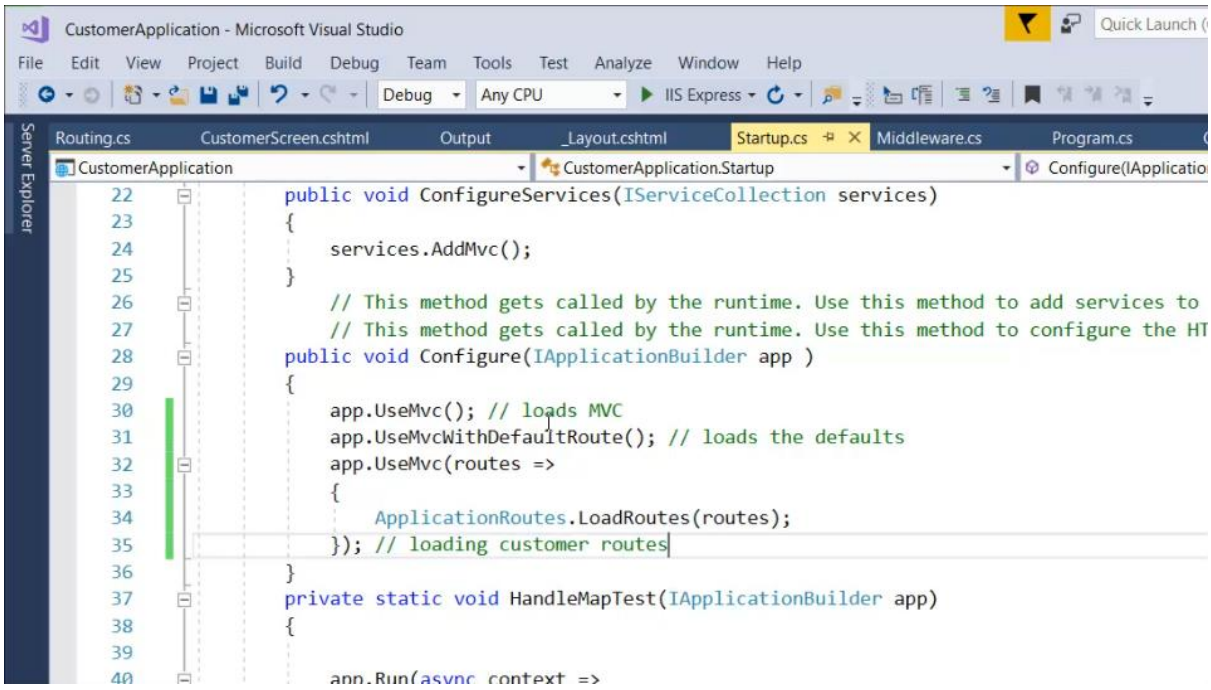

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
CustomerScreen.cshtml
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
CustomerController.cs
CustomerApplication
CustomerApplication.Startup
Configure(IApplicatio
}
public IConfiguration Configuration { get; }
public void ConfigureServices(IServiceCollection services)
{
services.AddMvc();
}
// This method gets called by the runtime. Use this method to add
// This method gets called by the runtime. Use this method to conf
public void Configure(IApplicationBuilder app)
{
app.UseStaticFiles(); // html
app.UseMvc();
app.UseMvcWithDefaultRoute();
app.UseMvc(routes =>
{
LoadRoutes(routes);
});
}
private static void HandleMapTest(IApplicationBuilder app)
Solution Explorer
Search Solution E
Server Explorer

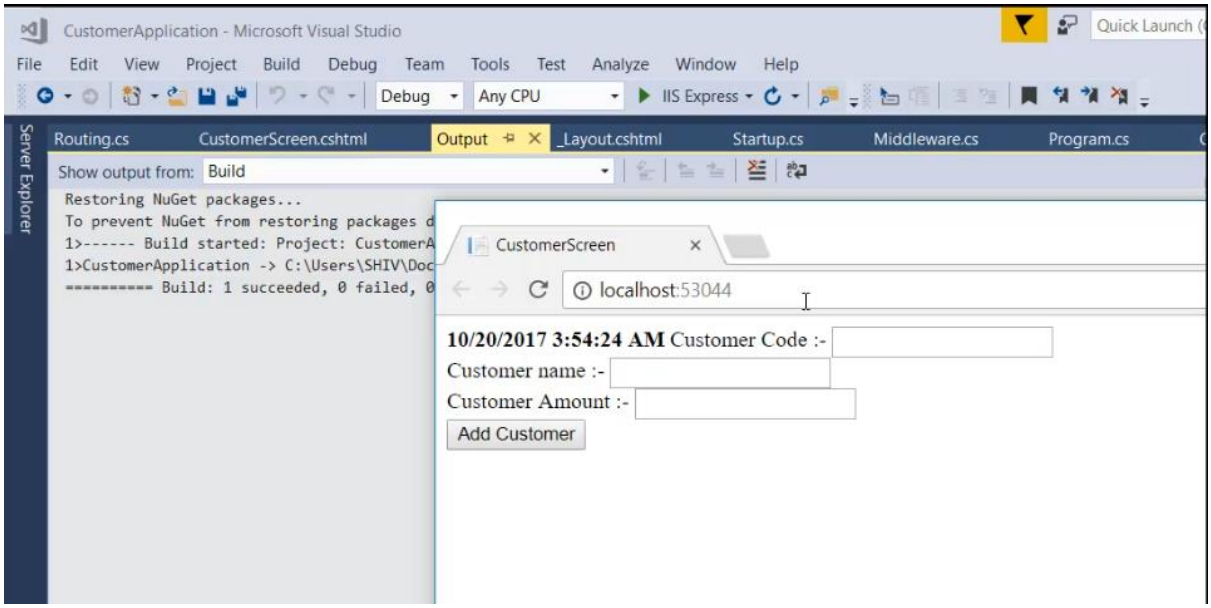


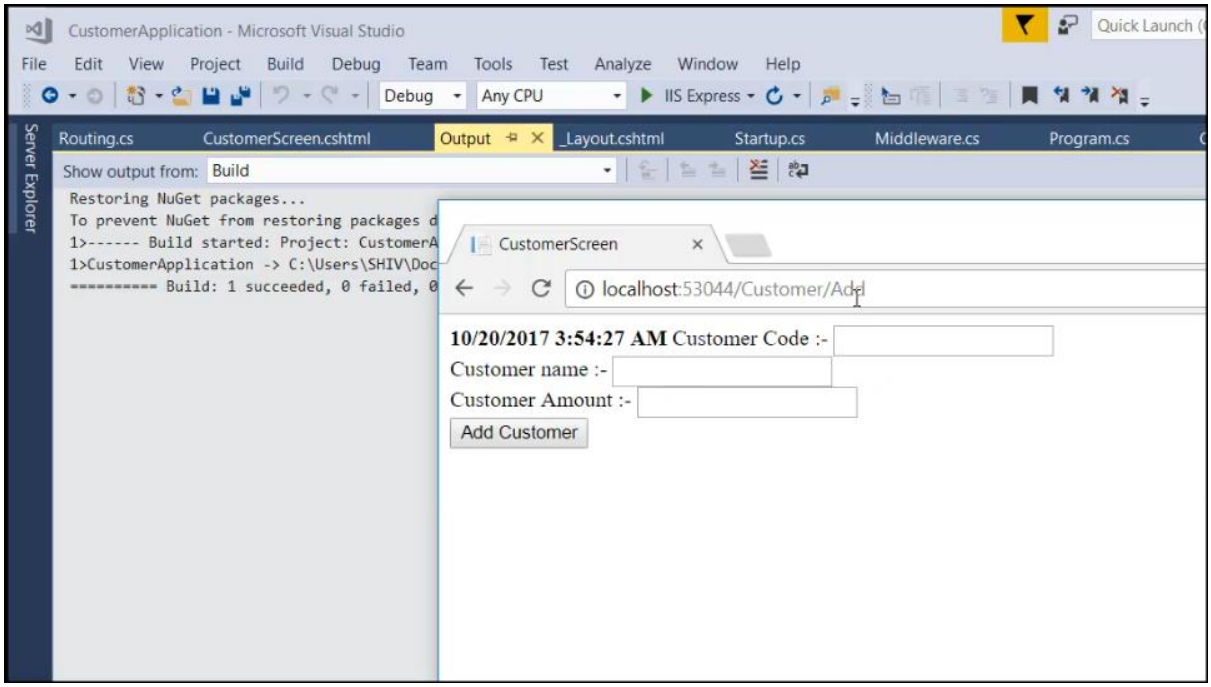
Now you can also create routing in separate folder. Add Routing folder and, add static class , with any name which is not reserved.

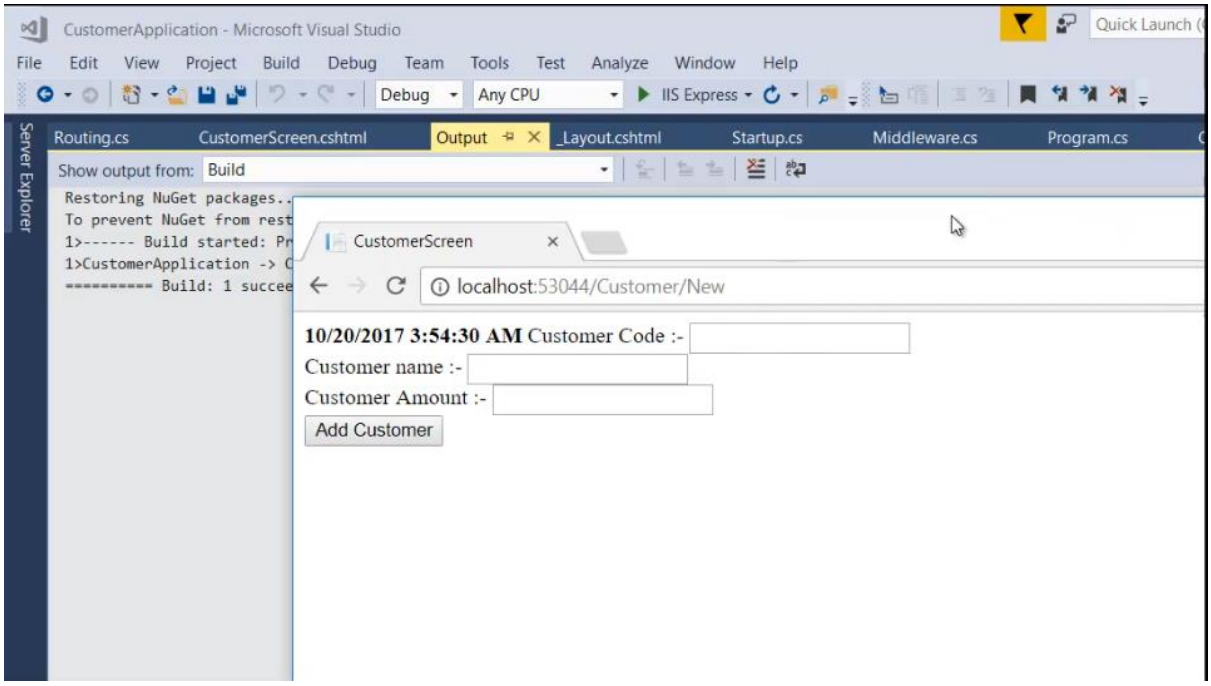


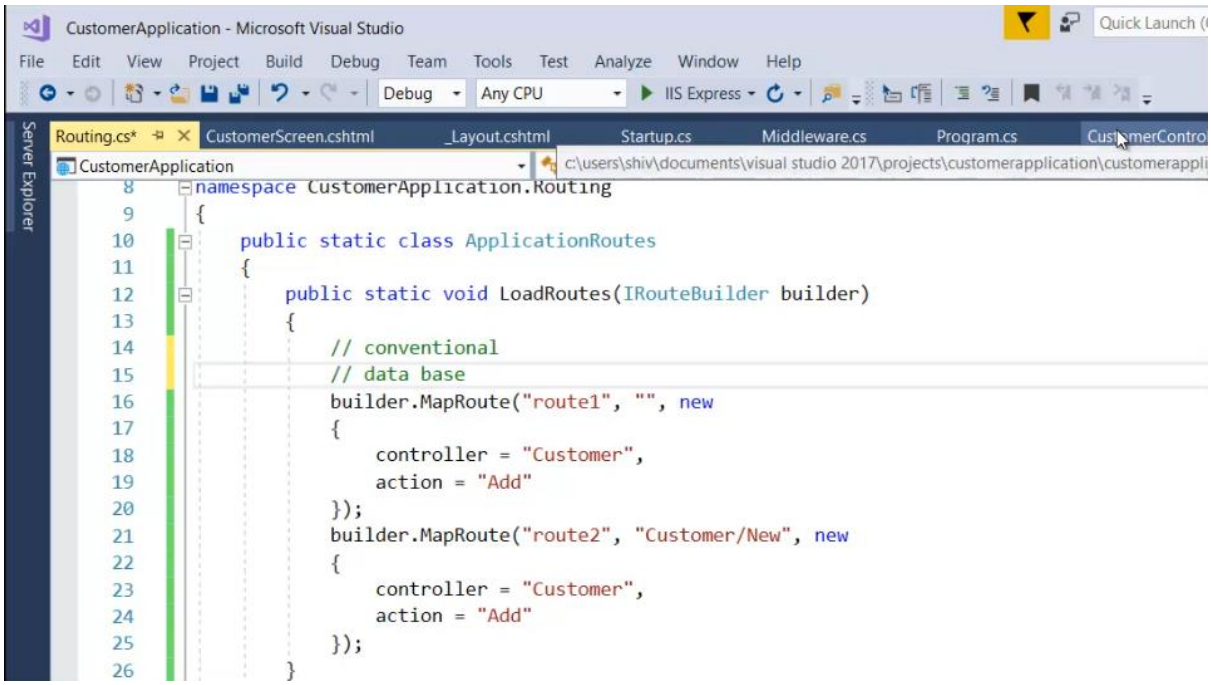

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Server Explorer
Routing.cs CustomerScreen.cshtml Output _Layout.cshtml Startup.cs Middleware.cs Program.cs
CustomerApplication
CustomerApplication.Routing.ApplicationRoutes
LoadRoutes(IRouteBu
7
8 namespace CustomerApplication.Routing
9 {
10 public static class ApplicationRoutes
11 {
12 public static void LoadRoutes(IRouteBuilder builder)
13 {
14 builder.MapRoute("route1", "", new
15 {
16 controller = "Customer",
17 action = "Add"
18 });
19 builder.MapRoute("route2", "Customer/New", new
20 {
21 controller = "Customer",
22 action = "Add"
23 });
24 }
25 }


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Server Explorer
Routing.cs CustomerScreen.cshtml Output _Layout.cshtml Startup.cs Middleware.cs Program.cs
CustomerApplication
CustomerApplication.Startup
Configure(IApplicatio
22 public void ConfigureServices(IServiceCollection services)
23 {
24 services.AddMvc();
25 }
26 // This method gets called by the runtime. Use this method to add services to
27 // This method gets called by the runtime. Use this method to configure the HT
28 public void Configure(IApplicationBuilder app)
29 {
30 app.UseMvc(); // loads MVC
31 app.UseMvcWithDefaultRoute(); // loads the defaults
32 app.UseMvc(routes =>
33 {
34 ApplicationRoutes.LoadRoutes(routes);
35 }); // loading customer routes
36 }
37 private static void HandleMapTest(IApplicationBuilder app)
38 {
39
40 app.Run(async context =>

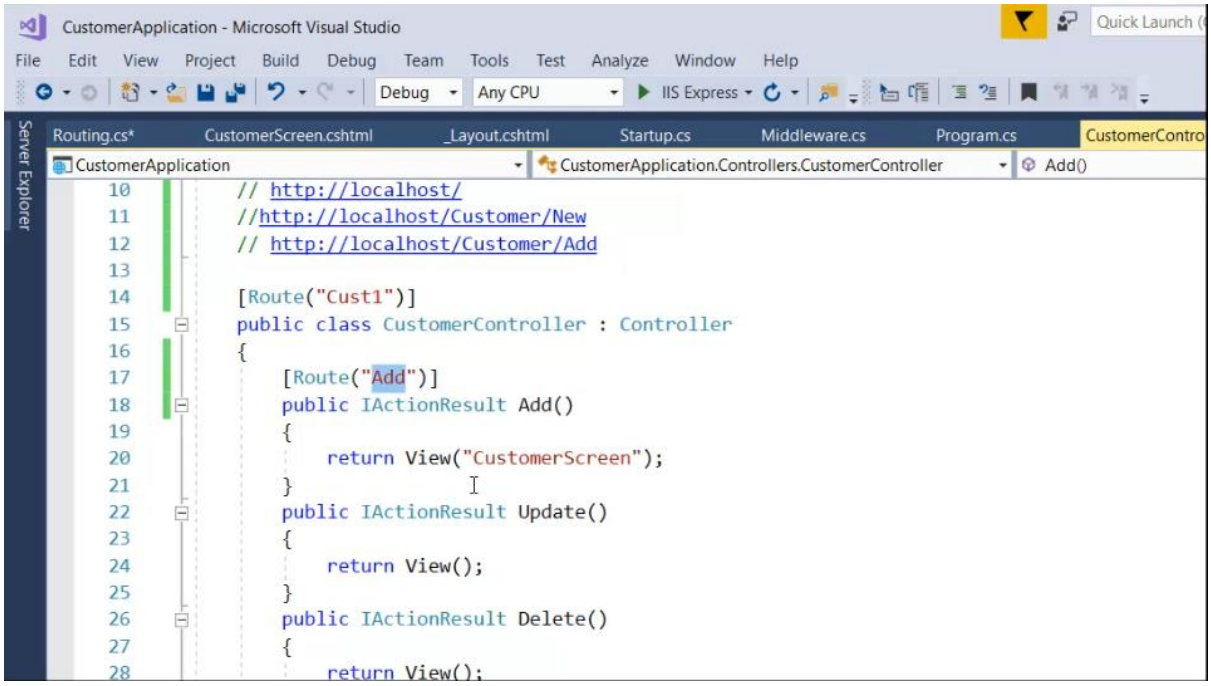

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Routing.cs
CustomerScreen.cshtml
Output
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
Show output from: Build
Restoring NuGet packages...
To prevent NuGet from restoring packages d
1>------ Build started: Project: CustomerA
1>CustomerApplication -> C:\Users\SHIV\Doc
========== Build: 1 succeeded, 0 failed, 0
CustomerScreen
localhost:53044
10/20/2017 3:54:24 AM Customer Code :-
Customer name :-
Customer Amount :-
Add Customer

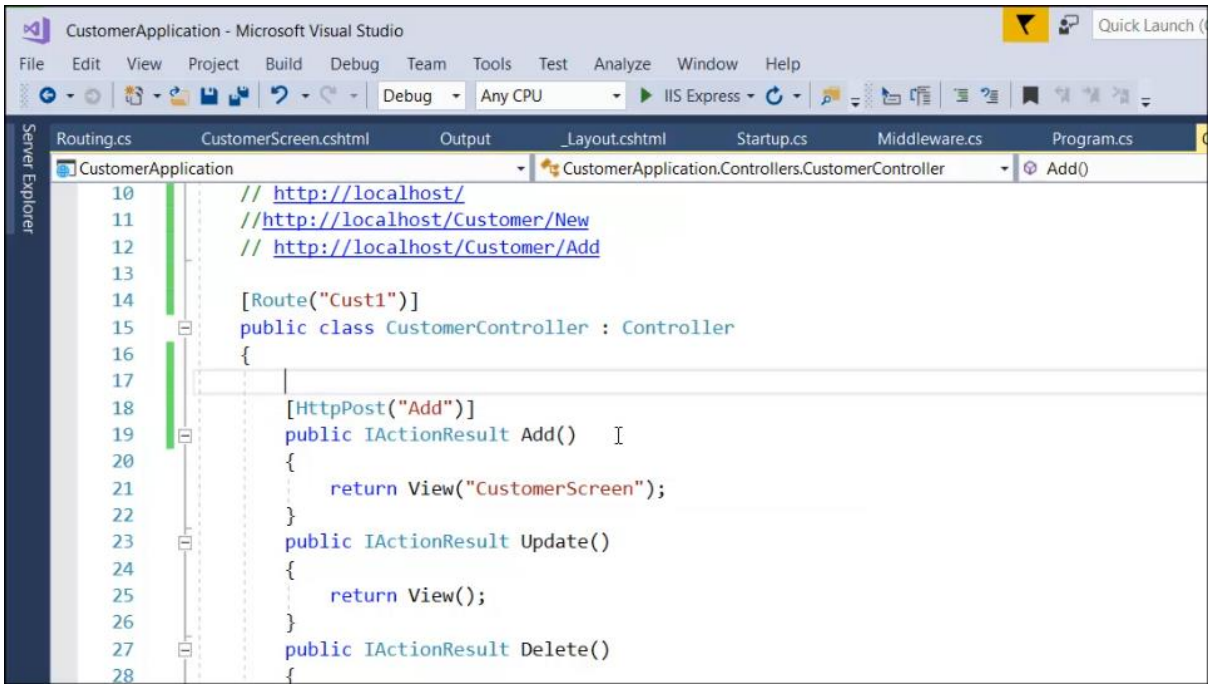

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Routing.cs
CustomerScreen.cshtml
Output
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
Show output from: Build
Restoring NuGet packages...
To prevent NuGet from restoring packages d
1>------ Build started: Project: CustomerA
1>CustomerApplication -> C:\Users\SHIV\Doc
========== Build: 1 succeeded, 0 failed, 0
CustomerScreen
localhost:53044/Customer/Add
10/20/2017 3:54:27 AM Customer Code :-
Customer name :-
Customer Amount :-
Add Customer



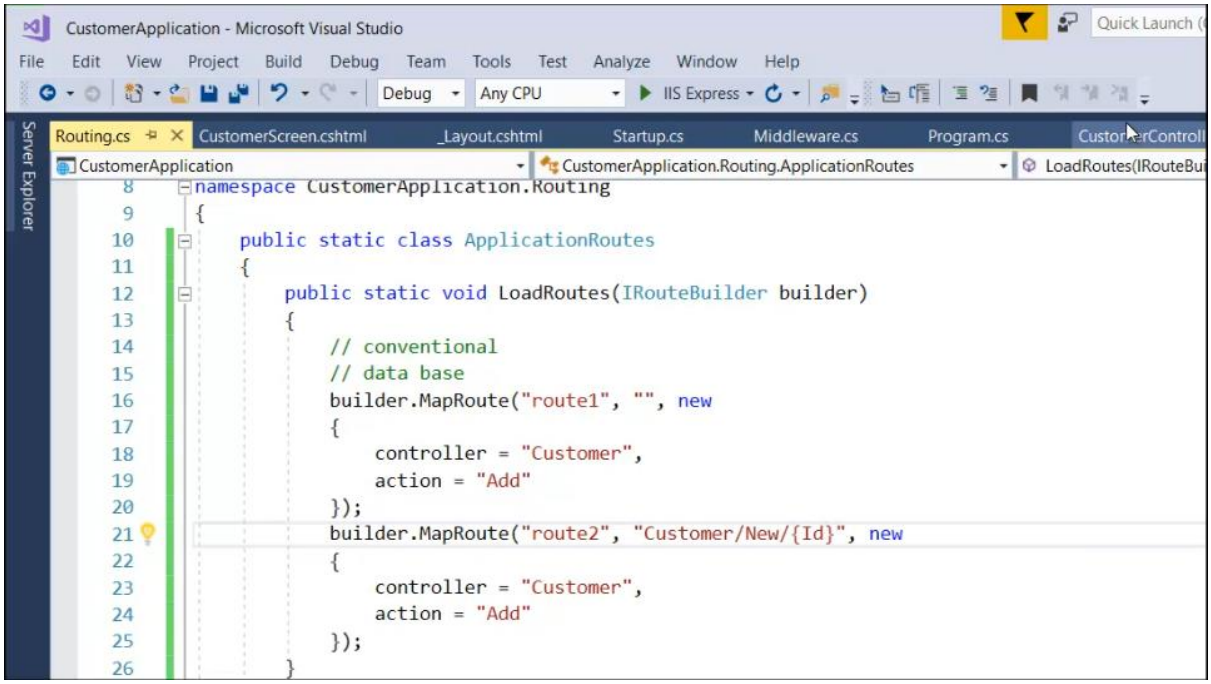


Attribute based routing

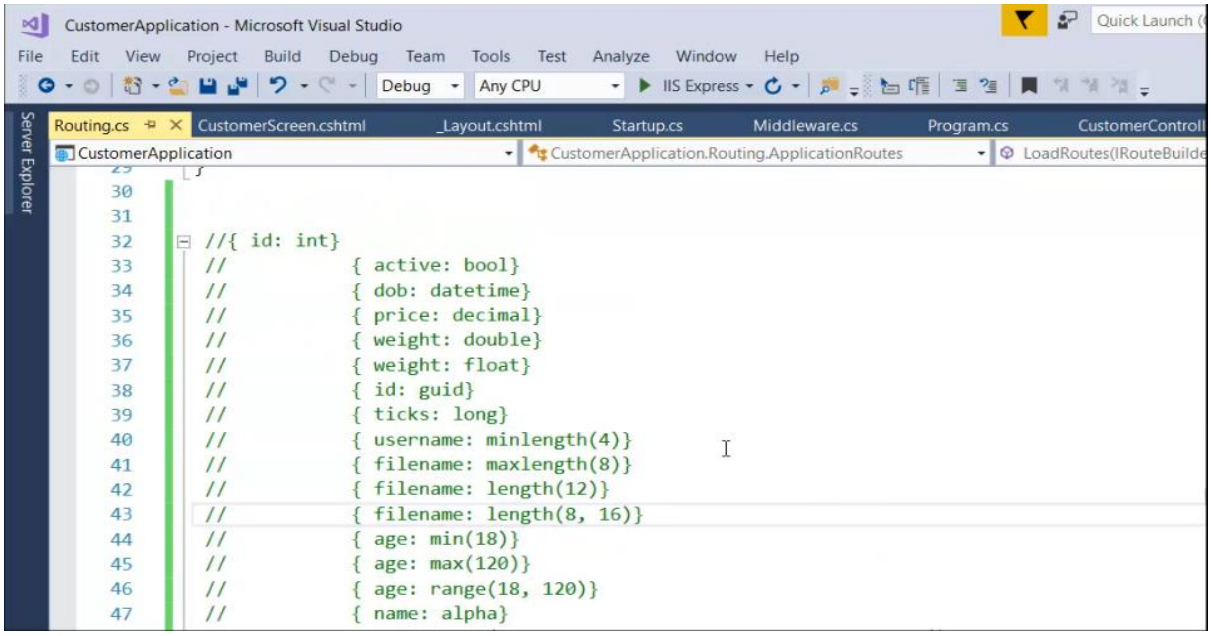


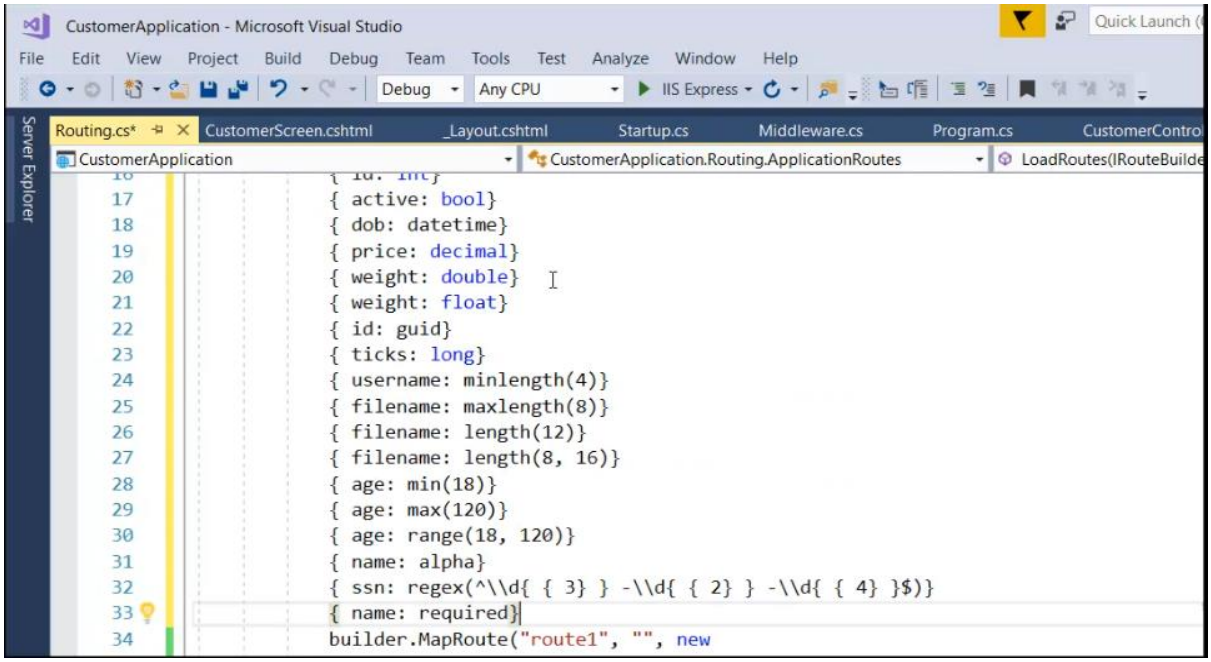


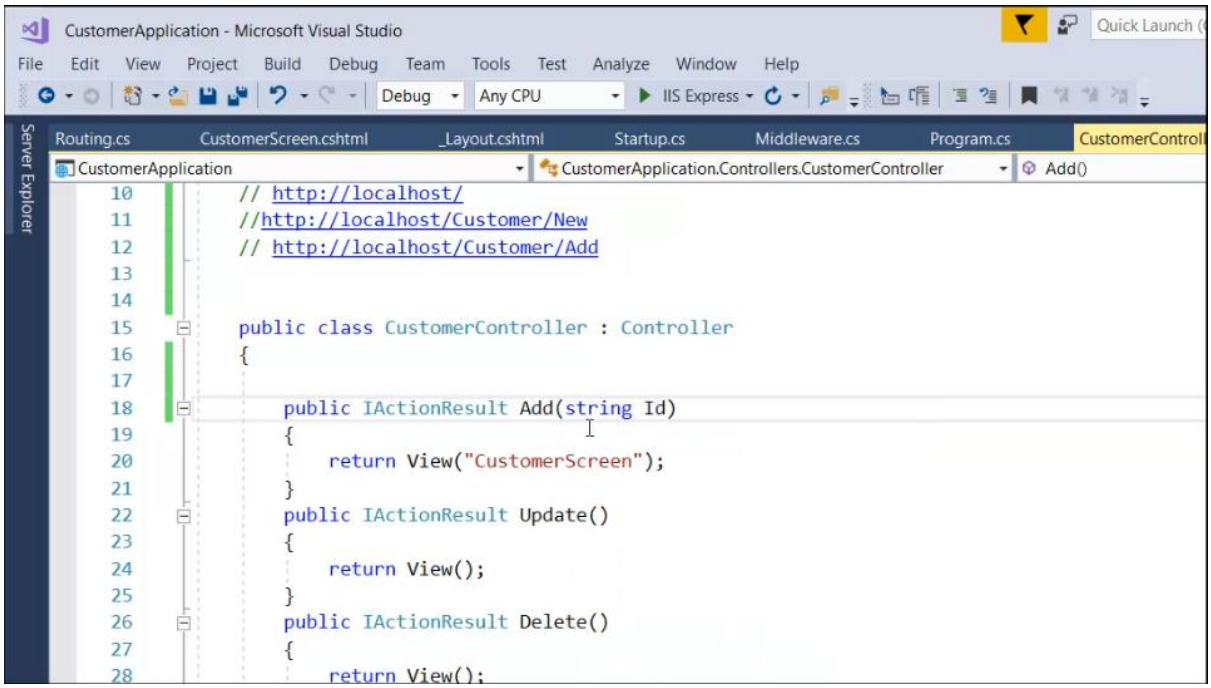
Note: Attribute based routing precedence over the conventional based routing. And attribute based you are not supposed to provide constraint.

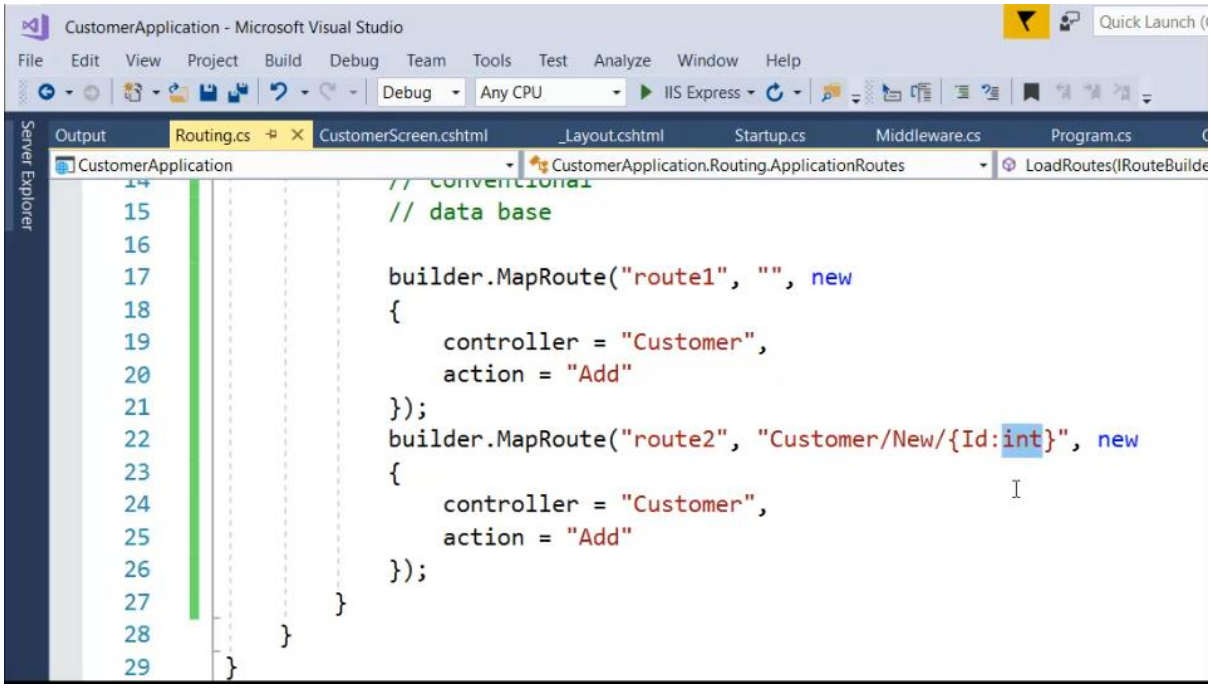


Validation over input parameter.

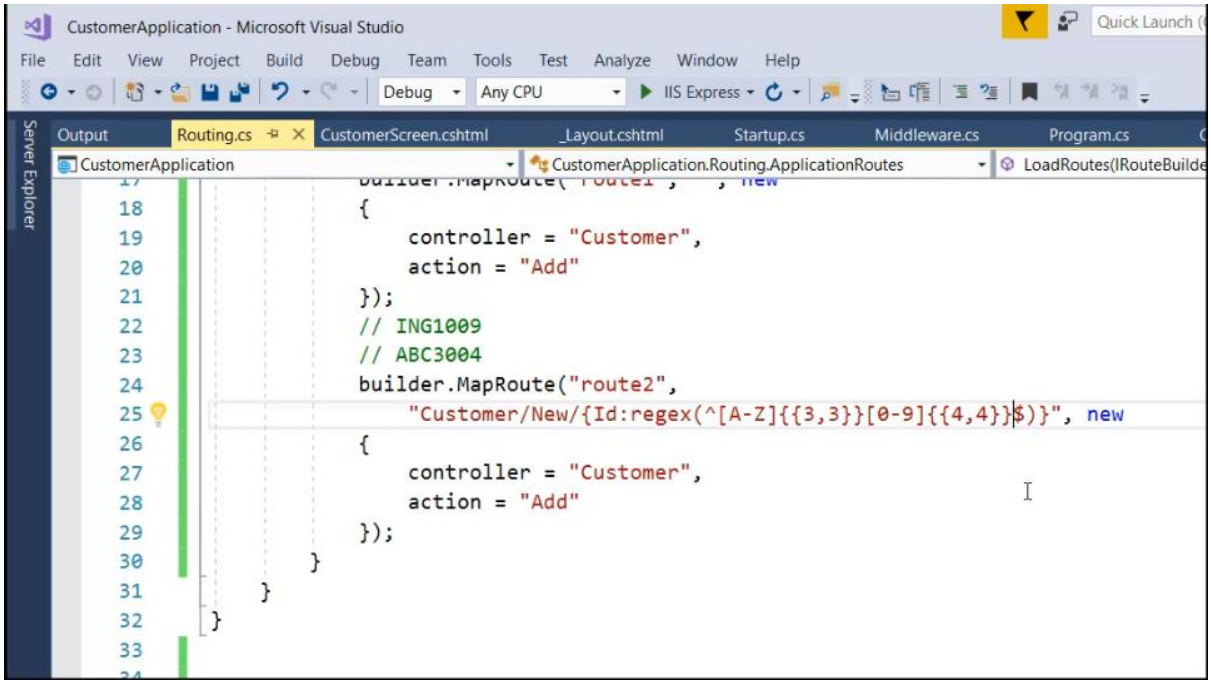


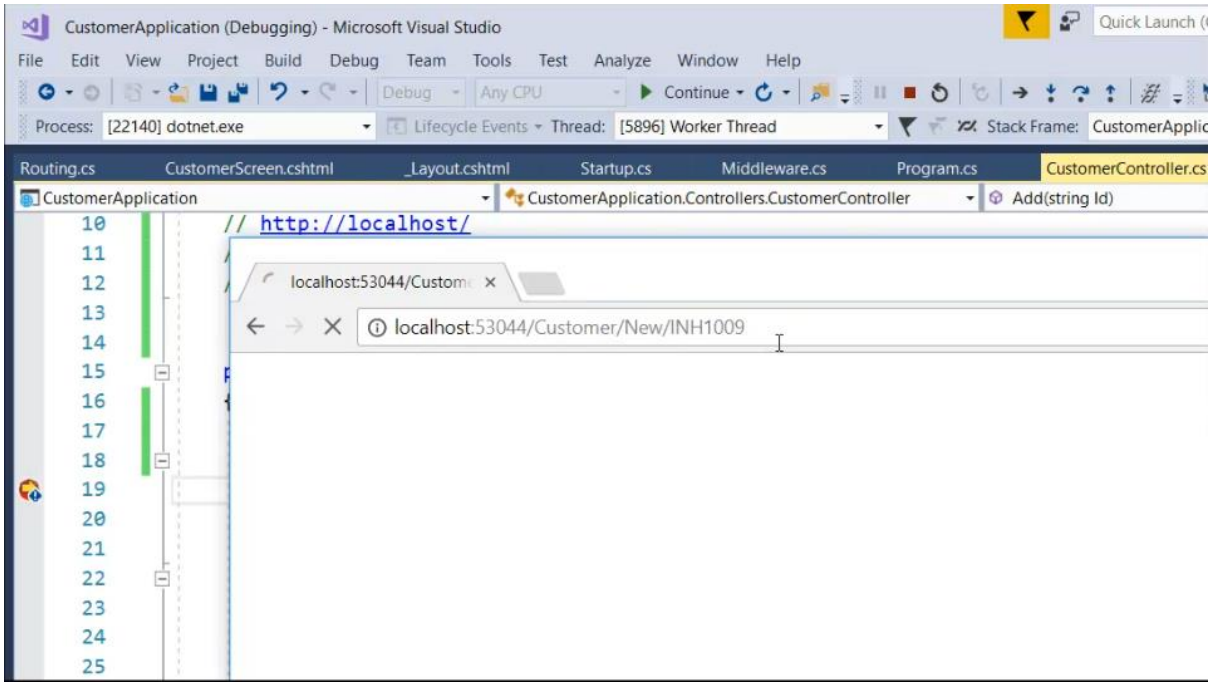

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Server Explorer
Routing.cs*
CustomerScreen.cshtml
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
CustomerApplication
CustomerApplication.Routing.ApplicationRoutes
LoadRoutes(IRouteBuilde
{ active: bool}
{ dob: datetime}
{ price: decimal}
{ weight: double}
{ weight: float}
{ id: guid}
{ ticks: long}
{ username: minlength(4)}
{ filename: maxlength(8)}
{ filename: length(12)}
{ filename: length(8, 16)}
{ age: min(18)}
{ age: max(120)}
{ age: range(18, 120)}
{ name: alpha}
{ ssn: regex(^\\d{ 3} -\\d{ 2} -\\d{ 4} $)}
{ name: required}
builder.MapRoute("route1", "", new

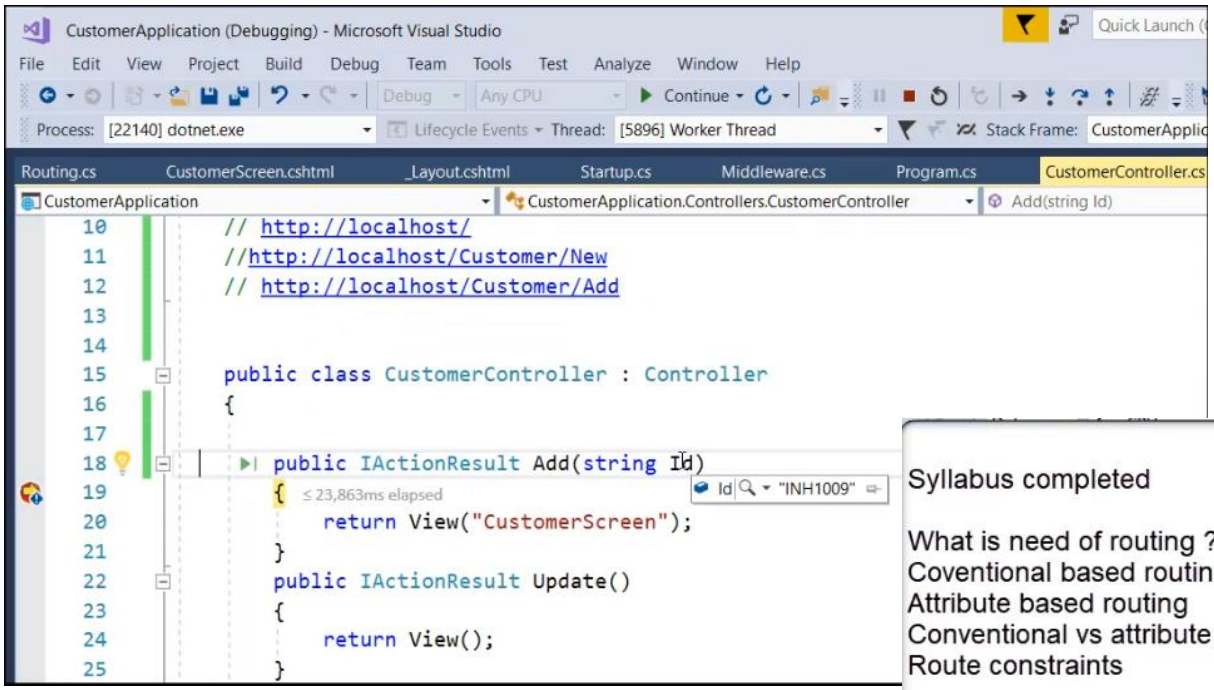

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Server Explorer
Routing.cs
CustomerScreen.cshtml
_Layout.cshtml
Startup.cs
Middleware.cs
Program.cs
CustomerApplication
CustomerApplication.Controllers.CustomerController
Add()
// http://localhost/
//http://localhost/Customer/New
// http://localhost/Customer/Add
public class CustomerController : Controller
{
public IActionResult Add(string Id)
{
return View("CustomerScreen");
}
public IActionResult Update()
{
return View();
}
public IActionResult Delete()
{
return View();


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Output Routing.cs CustomerScreen.cshtml _Layout.cshtml Startup.cs Middleware.cs Program.cs
CustomerApplication CustomerApplication.Routing.ApplicationRoutes LoadRoutes(IRouteBuilde
// data base
builder.MapRoute("route1", "", new
{
controller = "Customer",
action = "Add"
});
builder.MapRoute("route2", "Customer/New/{Id:int}", new
{
controller = "Customer",
action = "Add"
});
}
}
}

OR


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Output Routing.cs CustomerScreen.cshtml _Layout.cshtml Startup.cs Middleware.cs Program.cs
CustomerApplication CustomerApplication.Routing.ApplicationRoutes LoadRoutes(IRouteBuilde
{
controller = "Customer",
action = "Add"
});
// ING1009
// ABC3004
builder.MapRoute("route2",
"Customer/New/{Id:regex(^[A-Z]{{3,3}}[0-9]{{4,4}}$)}", new
{
controller = "Customer",
action = "Add"
});
}
}
}



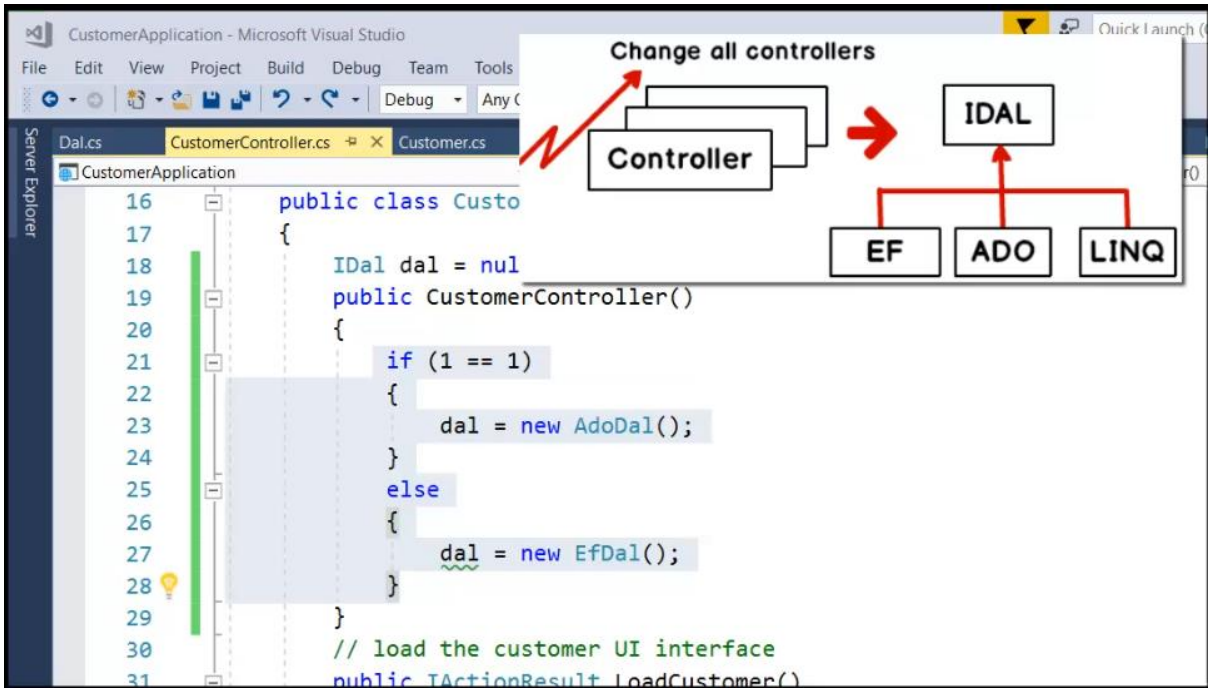


Syllabus completed

What is need of routing ?
Coventional based routing
Attribute based routing
Conventional vs attribute based
Route constraints

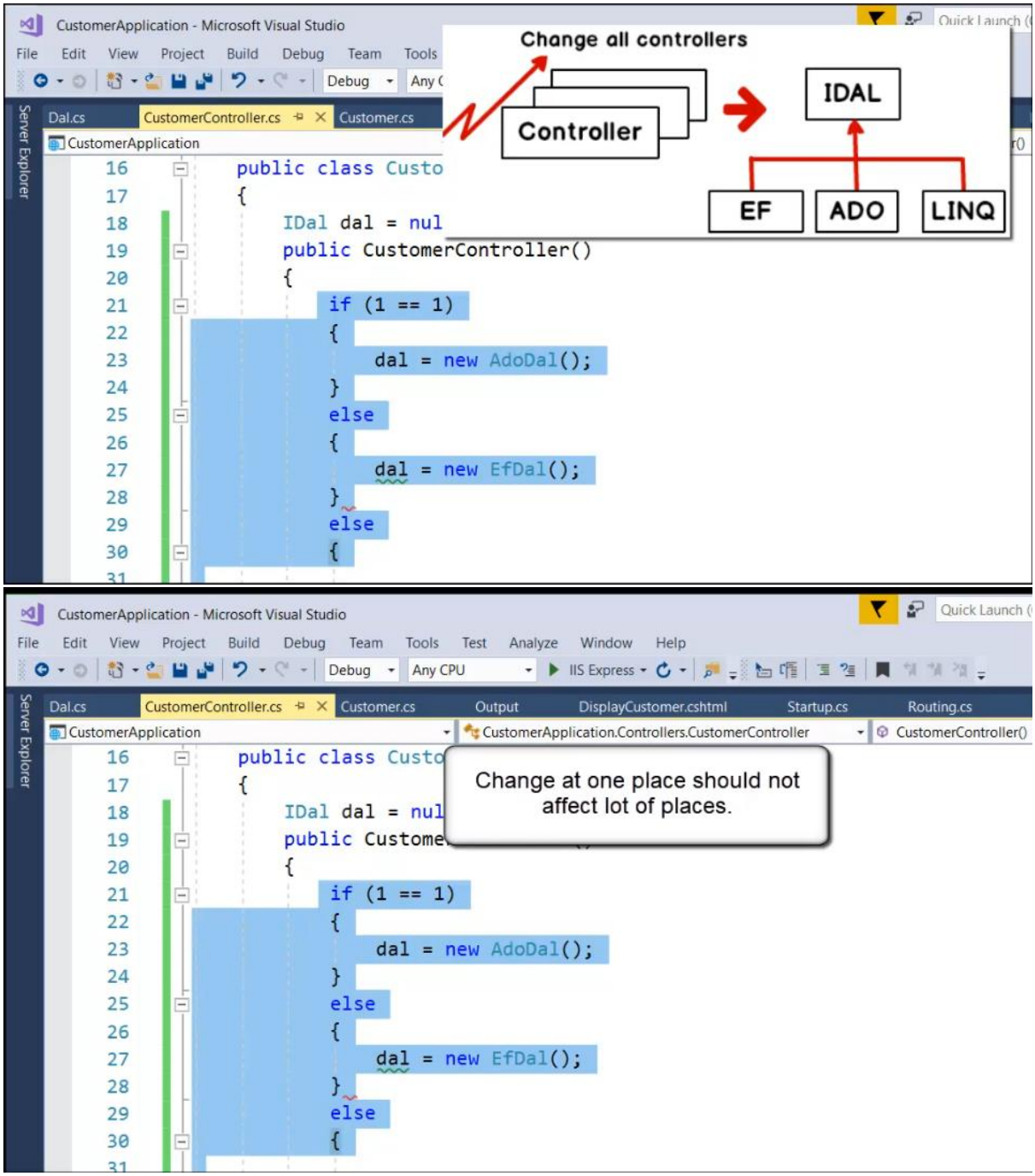
Depedency Injection (Scoped,Transient,Singleton & FromServ

DI , IOC
Scope,Trans,Single
FromServices

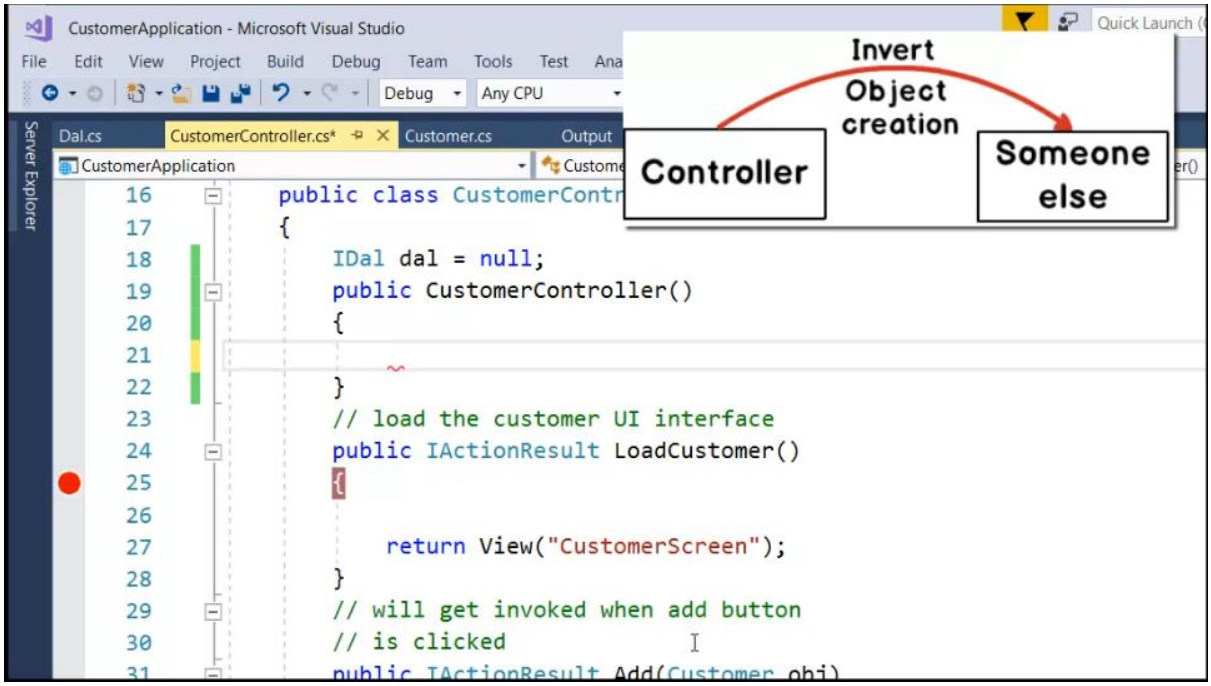


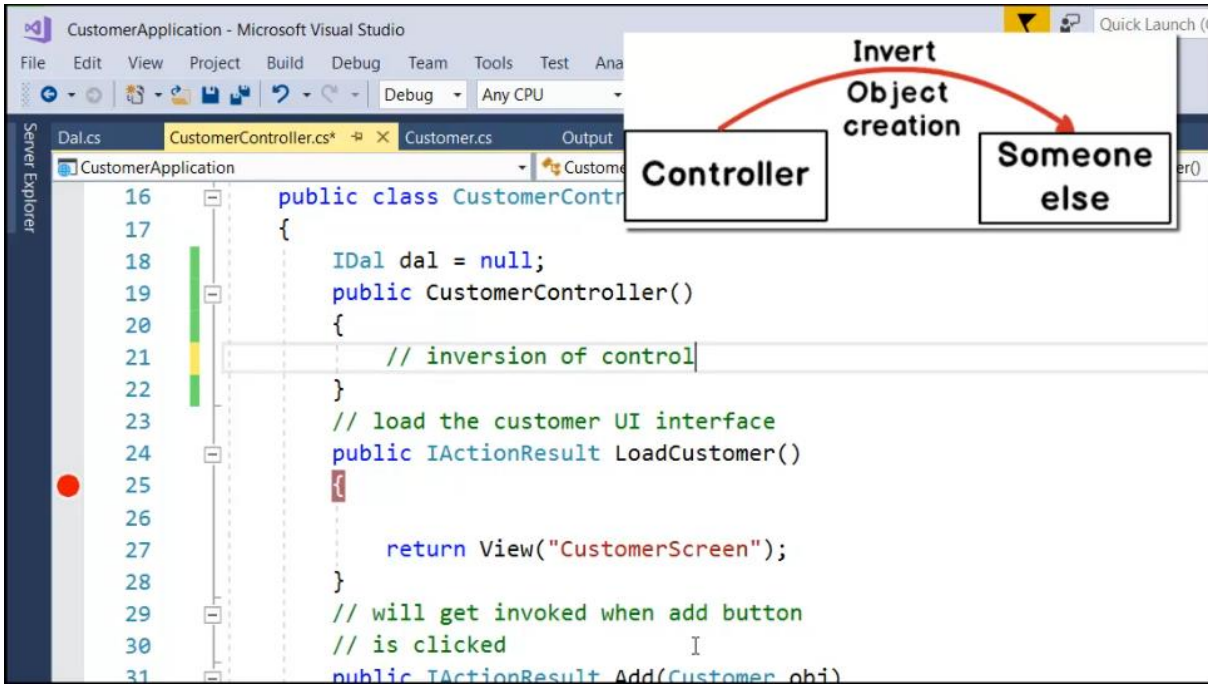
Now, suppose I have to use, one more dal class, LINQDAL, in that case I have change code in entire controller. SO we looking some type of mechanism, in which, where I will change at one place, and change replicate throughout your project.

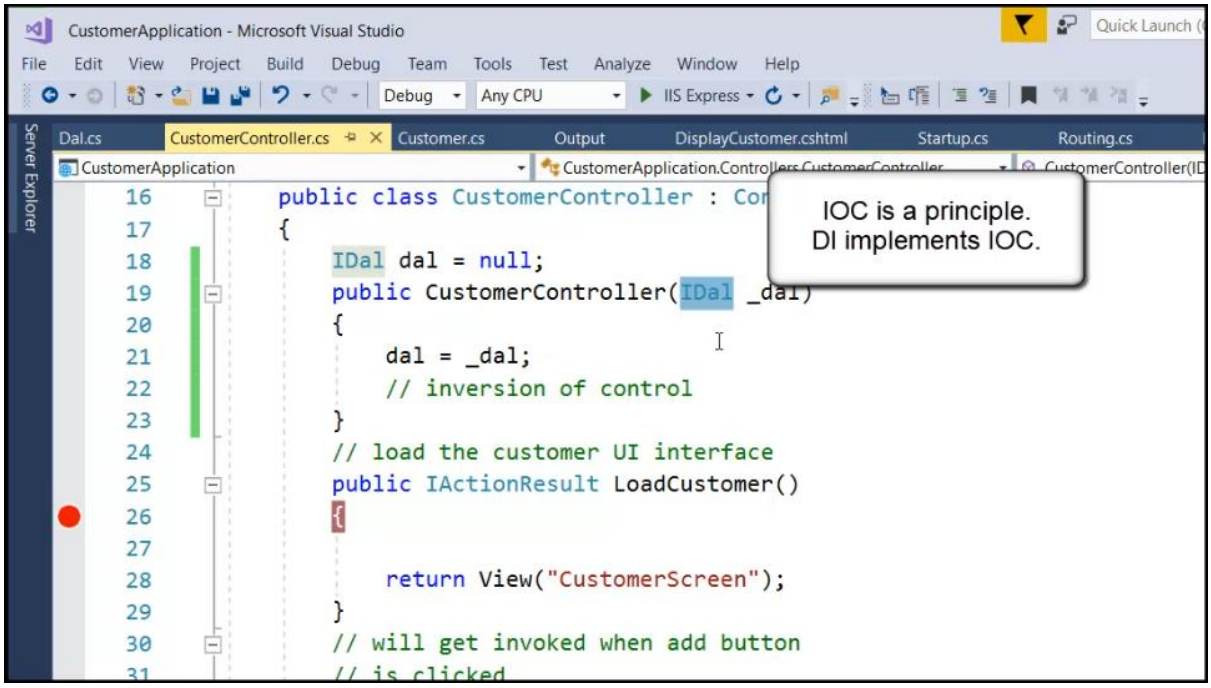
Sign of architect is that when you have made change in one place , it won't change in multiple places.



Controller is not appropriate place creating the of class. Because the work of the controller, is too taking the request, deciding the view.

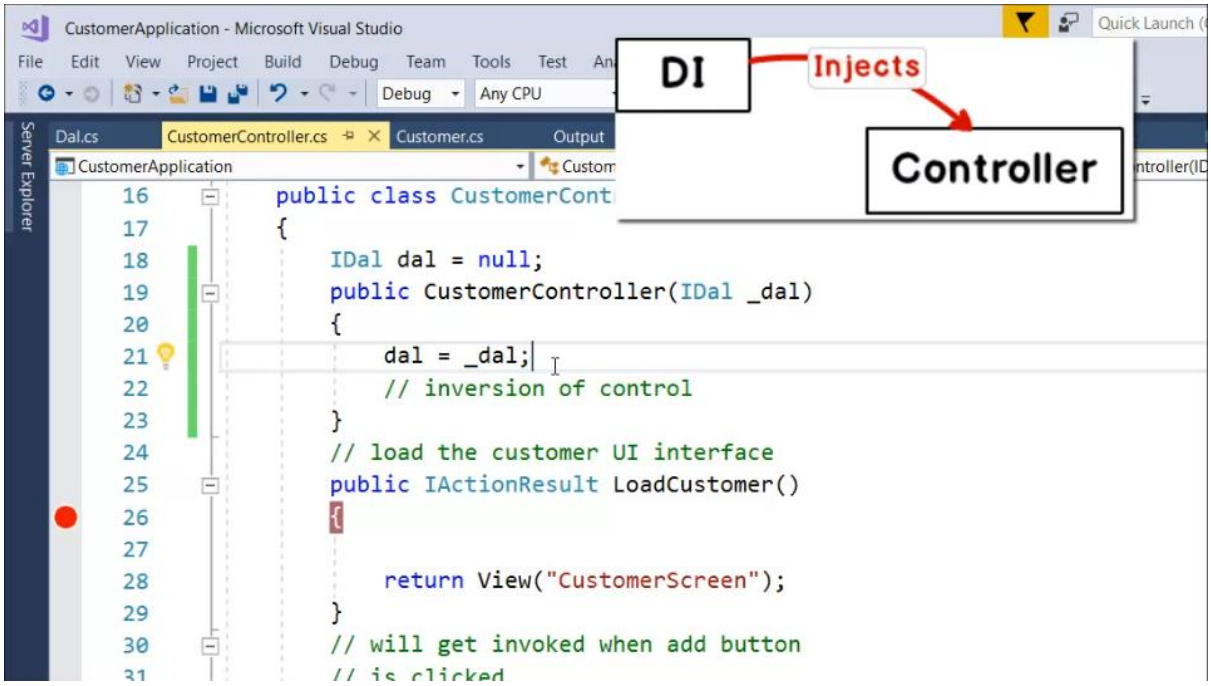

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Ana
Debug
Any CPU
Quick Launch
Server Explorer
Dal.cs
CustomerController.cs*
Customer.cs
Output
CustomerApplication
Invert
Object
creation
Controller
Someone
else
16 public class CustomerContr
17 {
18 IDal dal = null;
19 public CustomerController()
20 {
21
22 }
23 // load the customer UI interface
24 public IActionResult LoadCustomer()
25 {
26
27 return View("CustomerScreen");
28 }
29 // will get invoked when add button
30 // is clicked
31 public IActionResult Add(Customer obj)

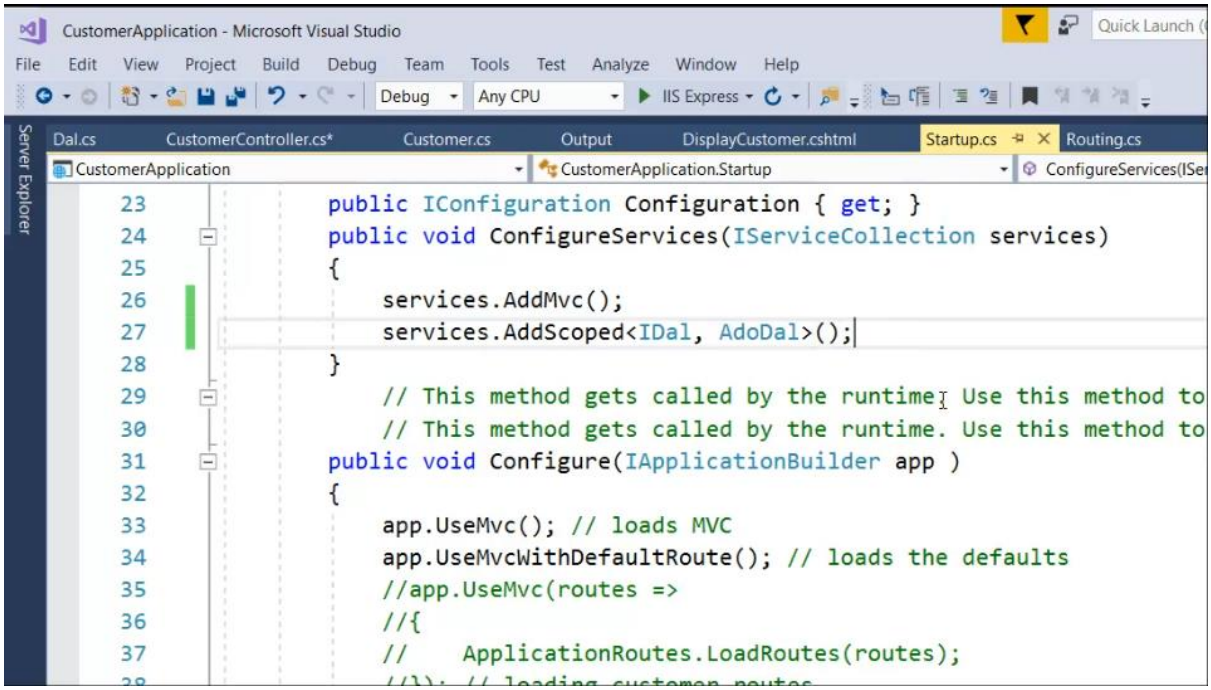

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Ana
Debug
Any CPU
Quick Launch
Server Explorer
Dal.cs
CustomerController.cs*
Customer.cs
Output
CustomerApplication
Invert
Object
creation
Controller
Someone
else
16 public class CustomerContr
17 {
18 IDal dal = null;
19 public CustomerController()
20 {
21 // inversion of control
22 }
23 // load the customer UI interface
24 public IActionResult LoadCustomer()
25 {
26
27 return View("CustomerScreen");
28 }
29 // will get invoked when add button
30 // is clicked
31 public IActionResult Add(Customer obj)

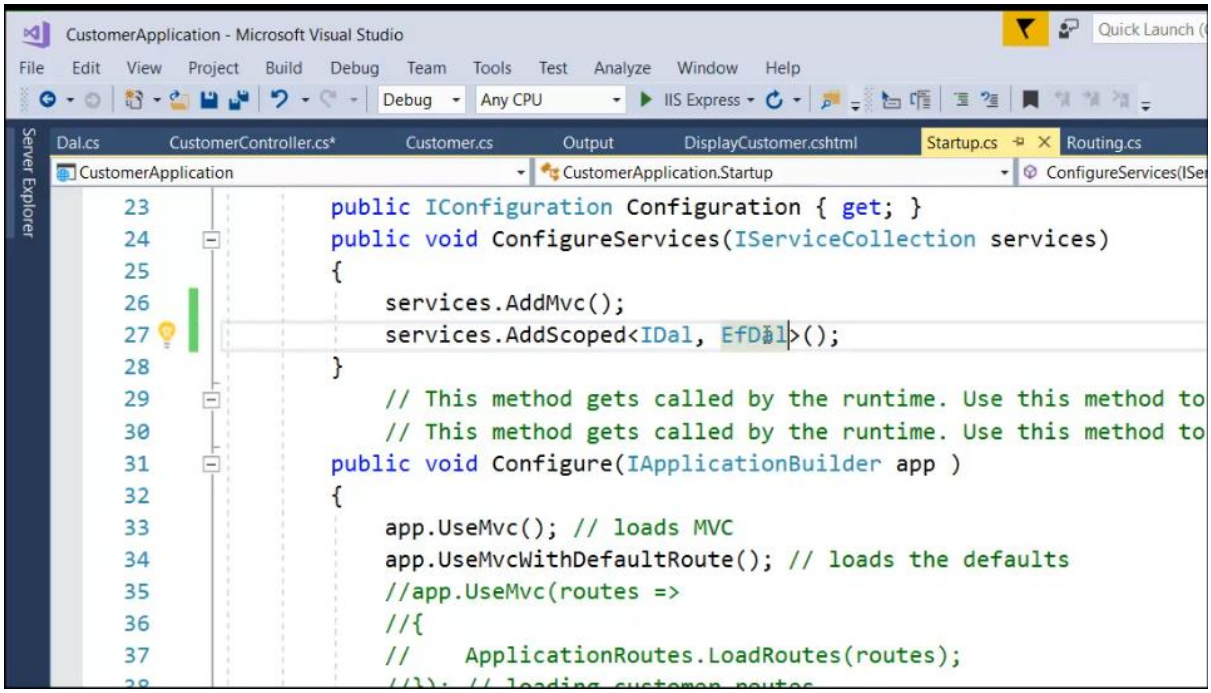


Note: ICO is a principle and DI is the way of implement IOC.

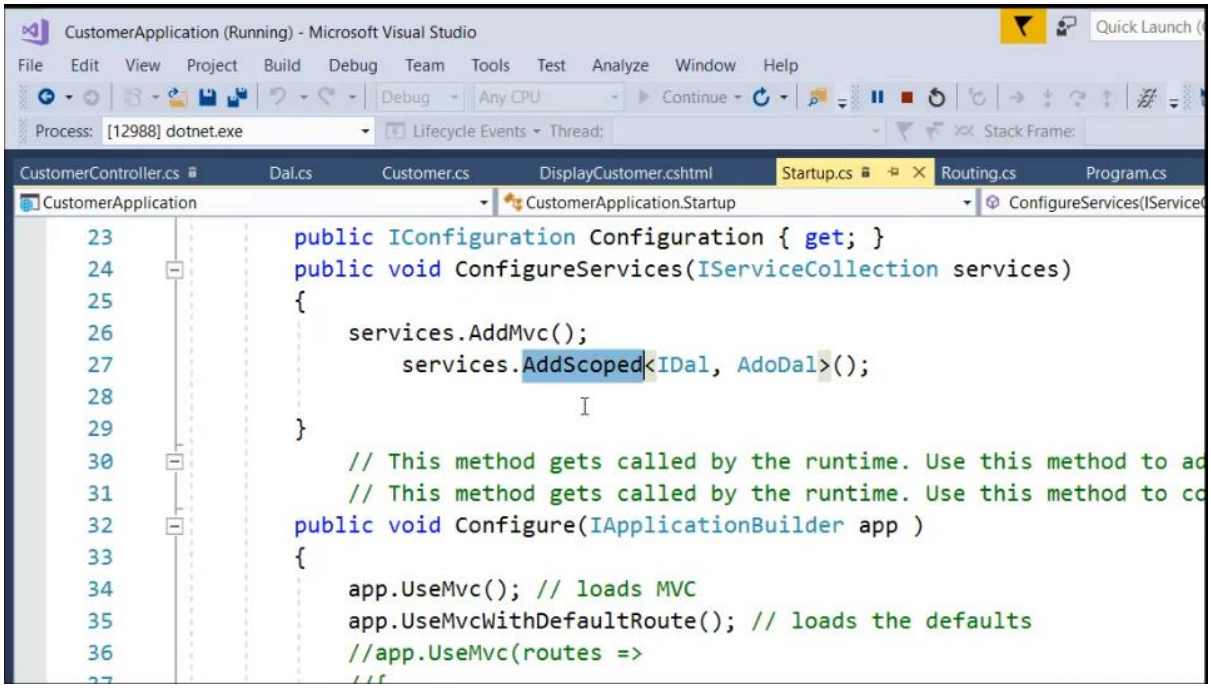
Now who decide to create object.

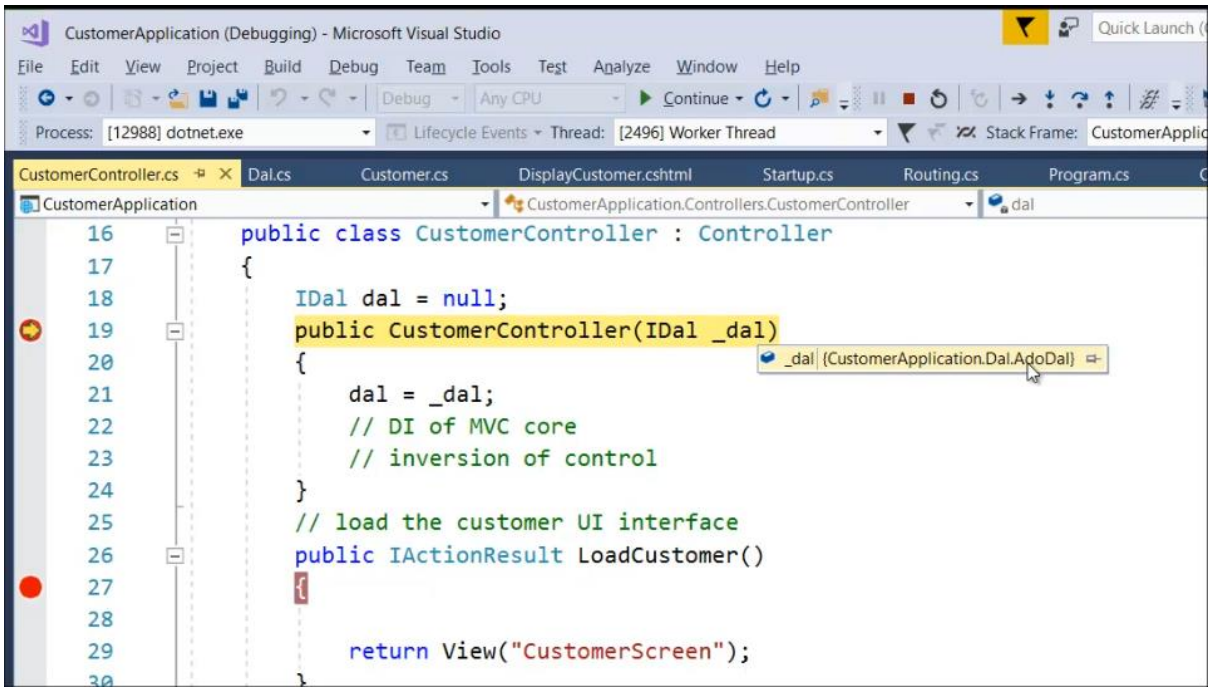




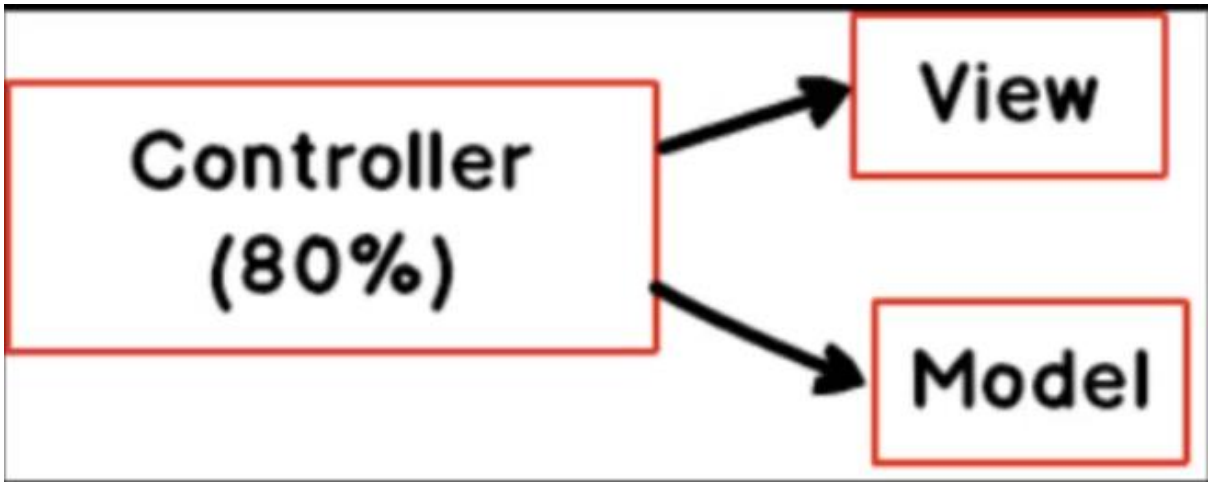


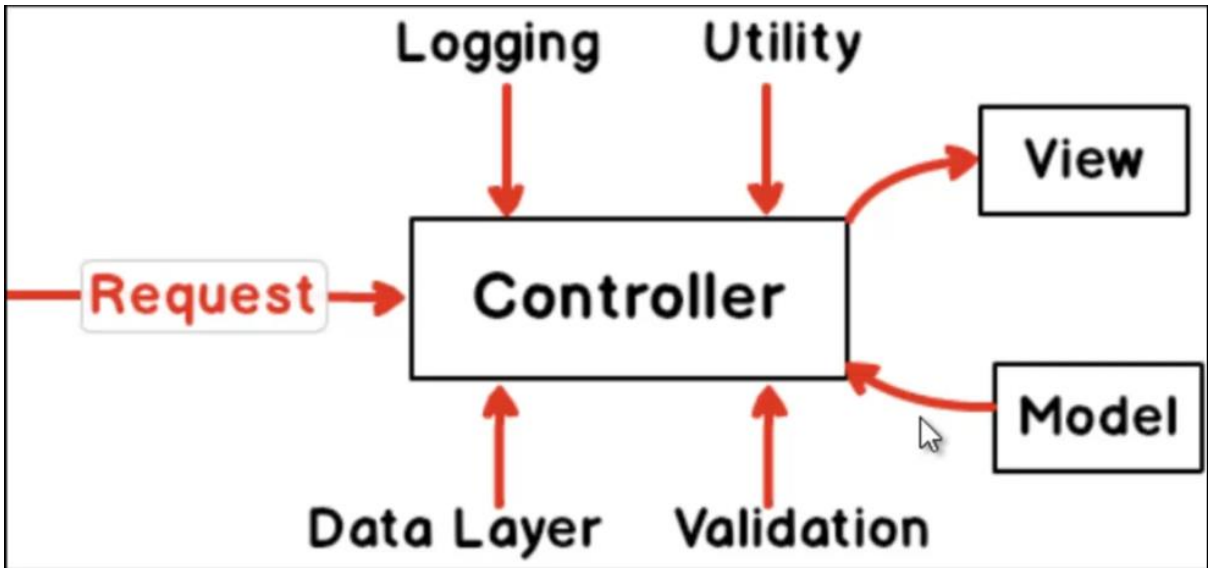
Note: Dependency injection, there is separate framework who actually injects the object and the client just have generic interface which to an interface, in the interface we will get actual concrete type.

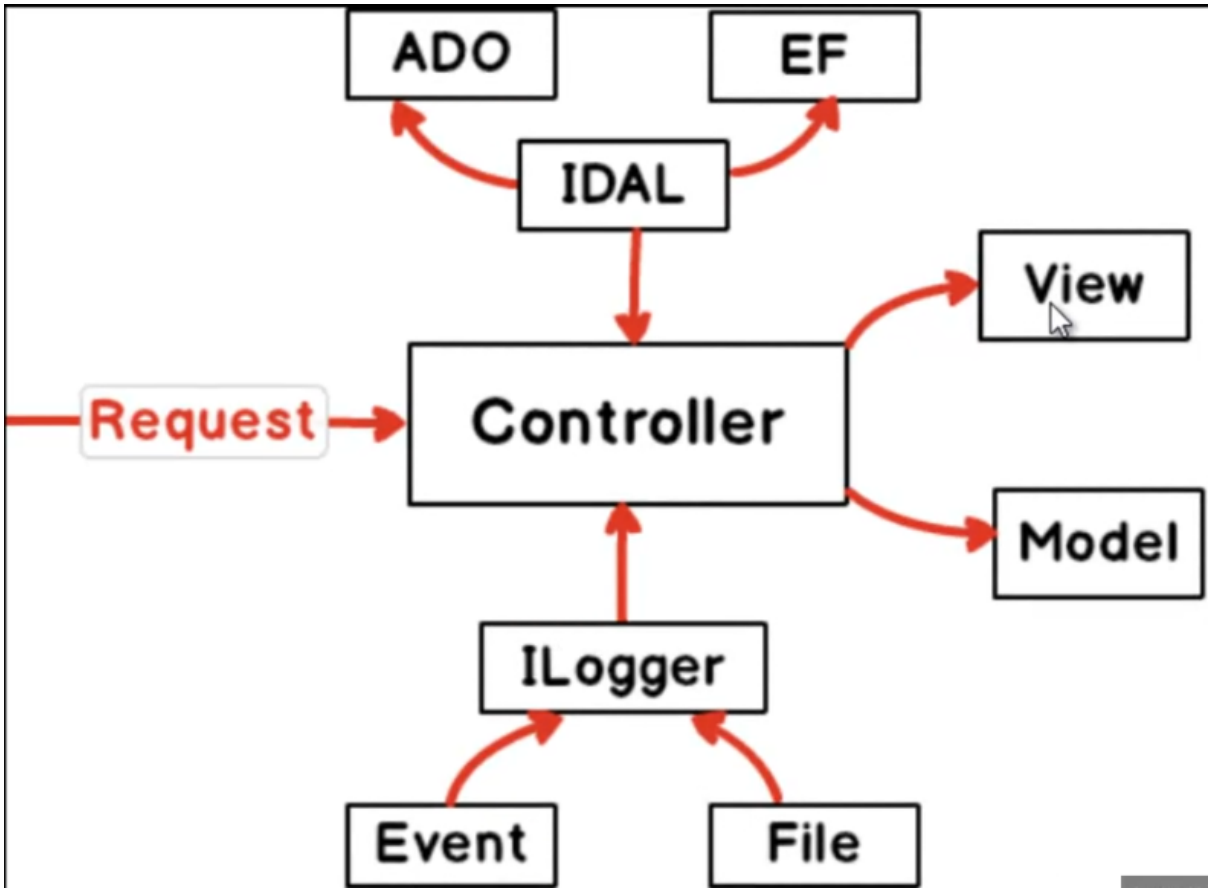


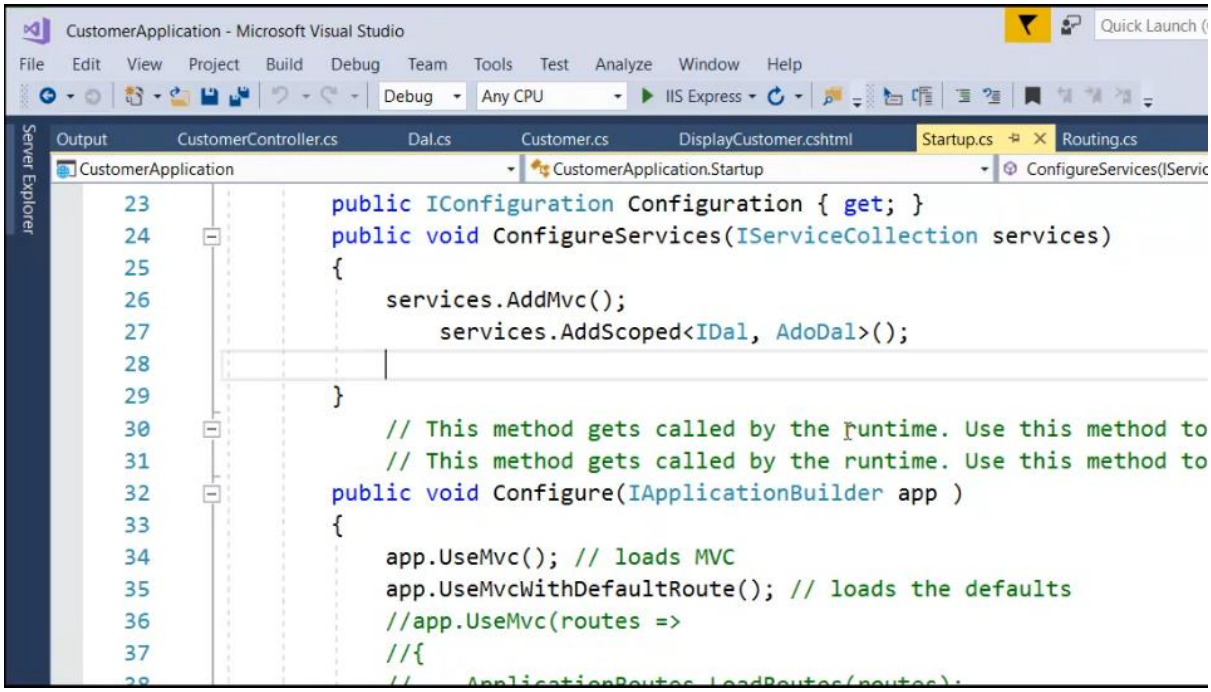


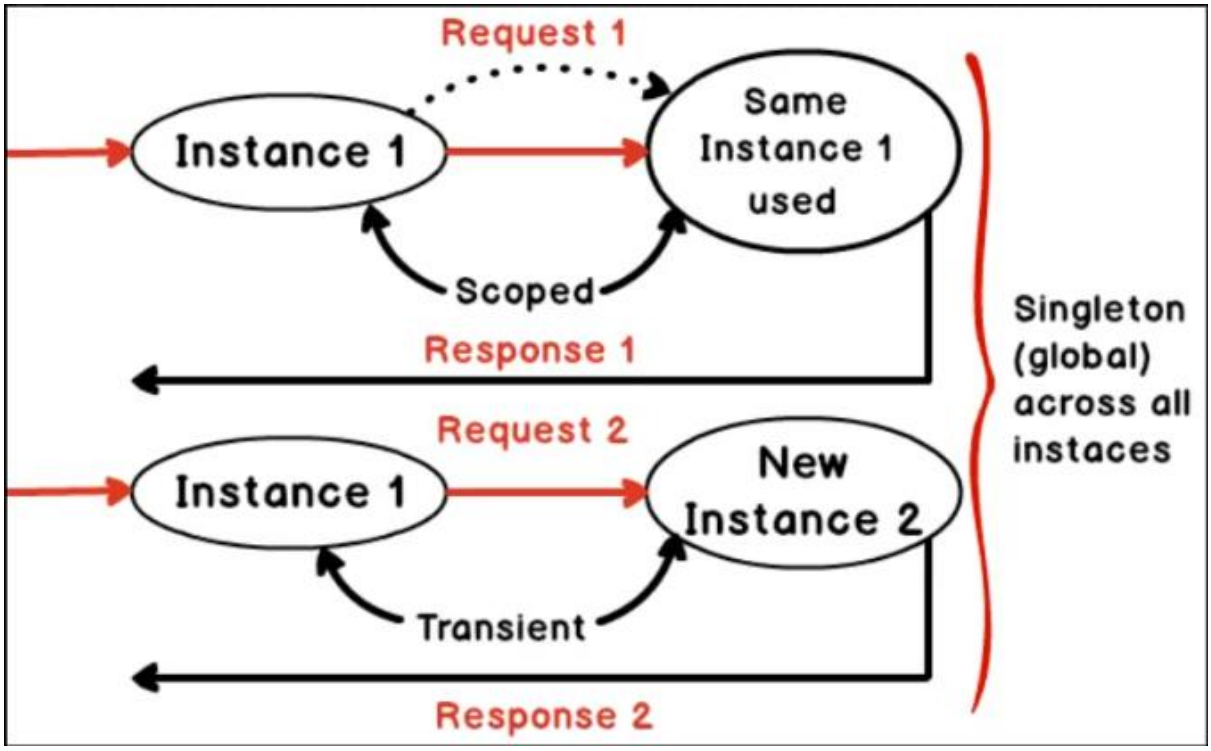
Note: Now customer controller doesn't do the decision, he has to inject ADODAL or EFDAL. It is done centrally with start-up file.


Controller
(80%)
View
Model

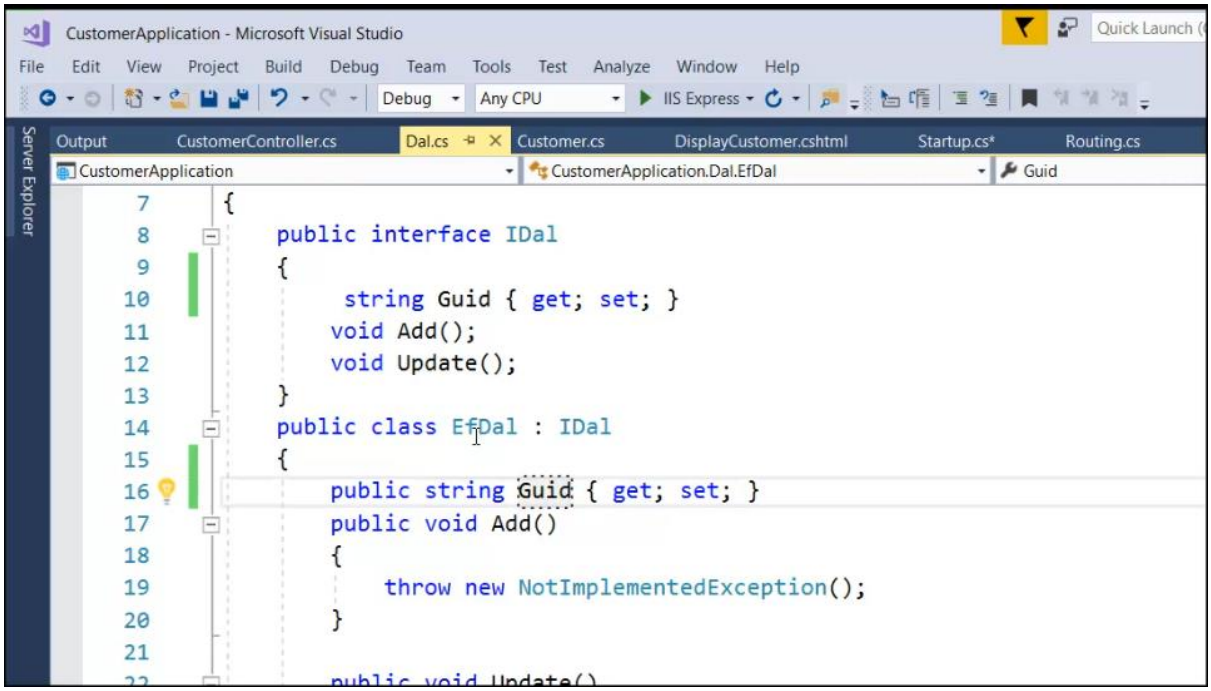

Logging
Utility
View
Request
Controller
Model
Data Layer
Validation

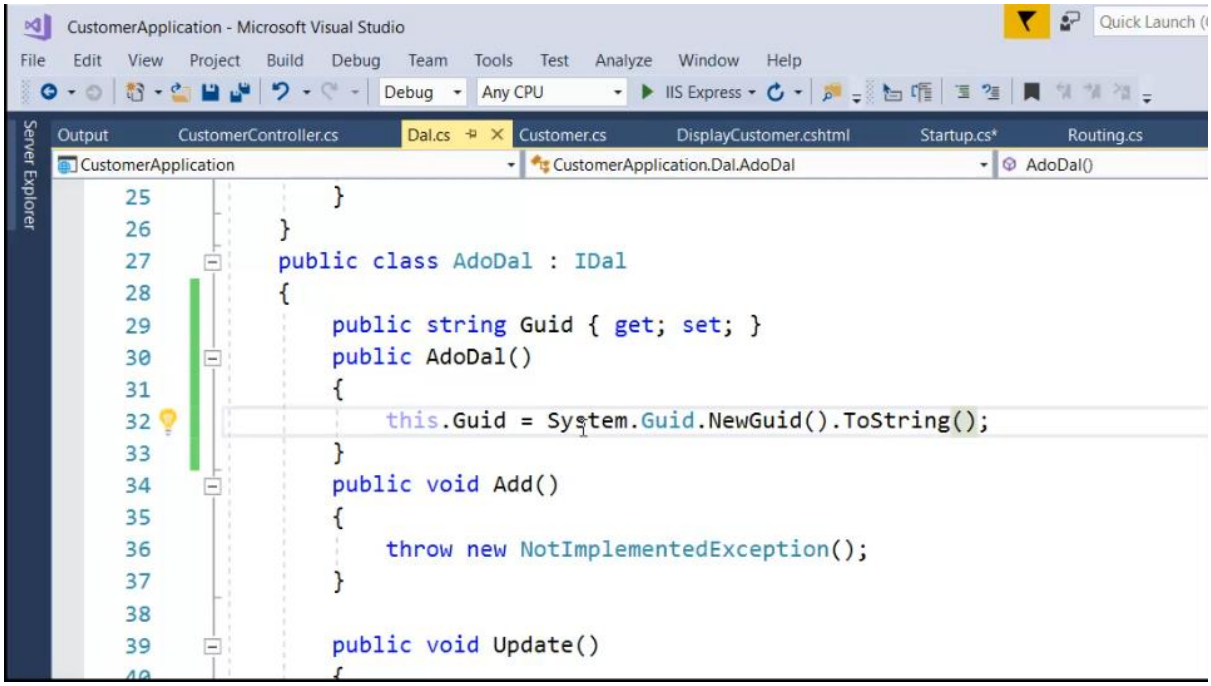

ADO
EF
IDAL
View
Request
Controller
Model
ILogger
Event
File

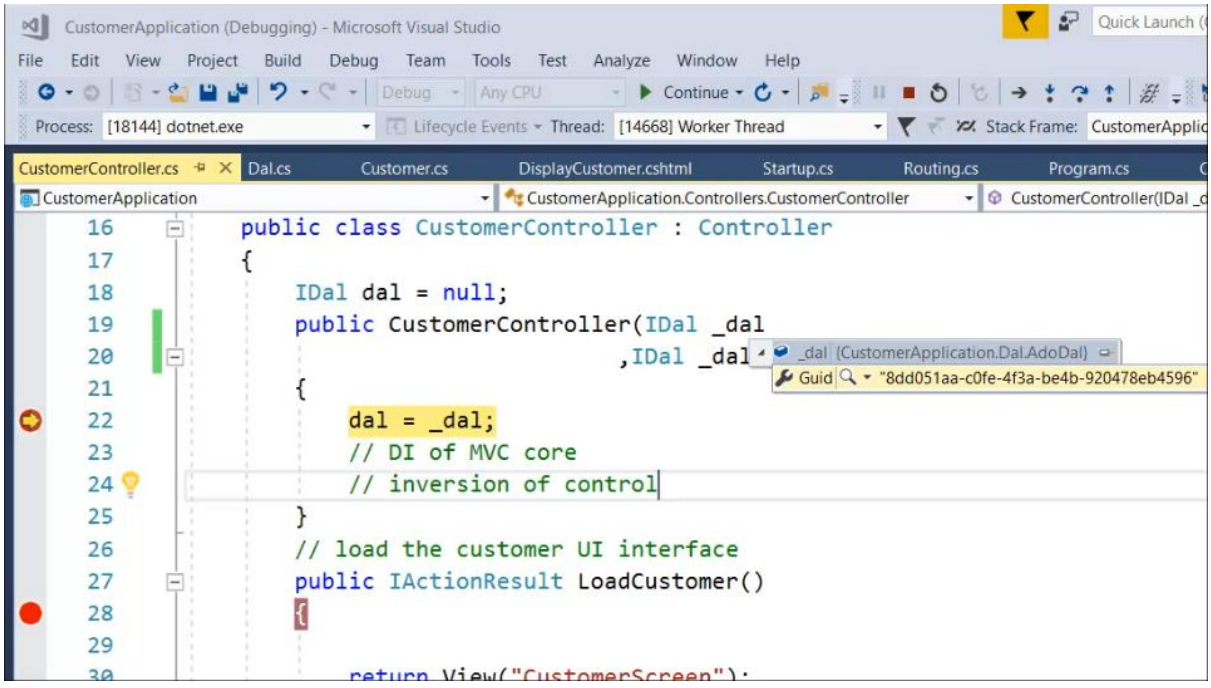

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Output
CustomerController.cs
Dal.cs
Customer.cs
DisplayCustomer.cshtml
Startup.cs
Routing.cs
CustomerApplication
CustomerApplication.Startup
ConfigureServices(IServic
Server Explorer
23 public IConfiguration Configuration { get; }
24 public void ConfigureServices(IServiceCollection services)
25 {
26 services.AddMvc();
27 services.AddScoped<IDal, AdoDal>();
28
29 }
30 // This method gets called by the runtime. Use this method to
31 // This method gets called by the runtime. Use this method to
32 public void Configure(IApplicationBuilder app)
33 {
34 app.UseMvc(); // loads MVC
35 app.UseMvcWithDefaultRoute(); // loads the defaults
36 //app.UseMvc(routes =>
37 //{

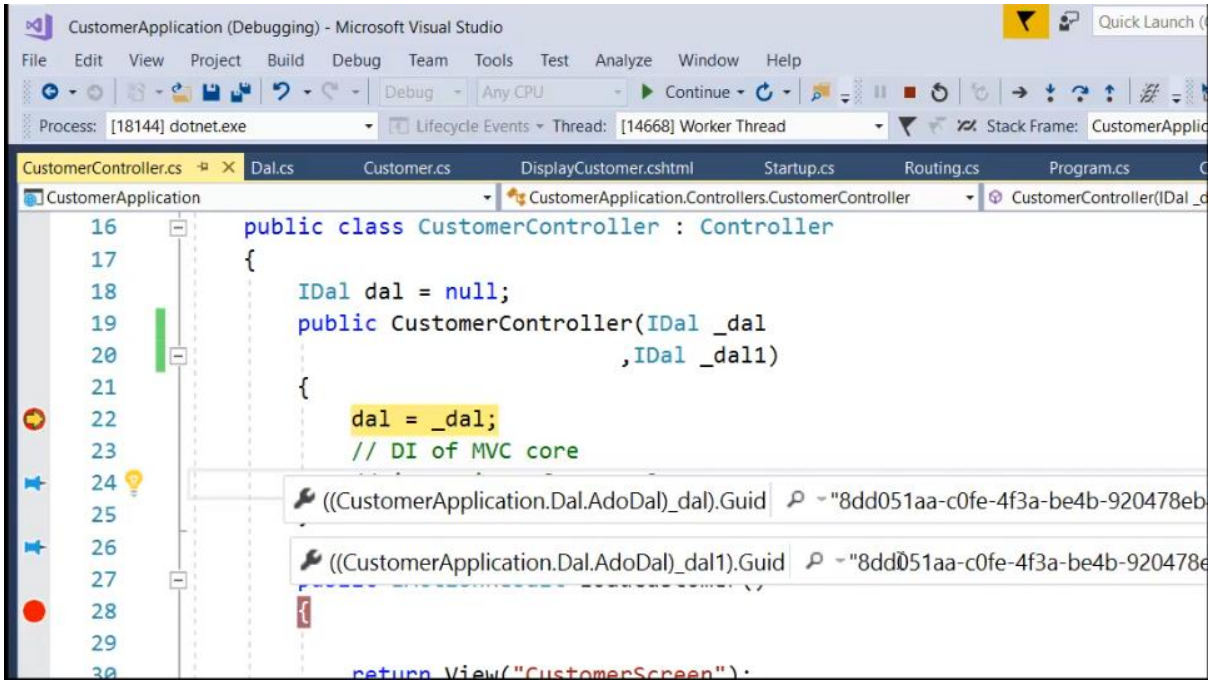


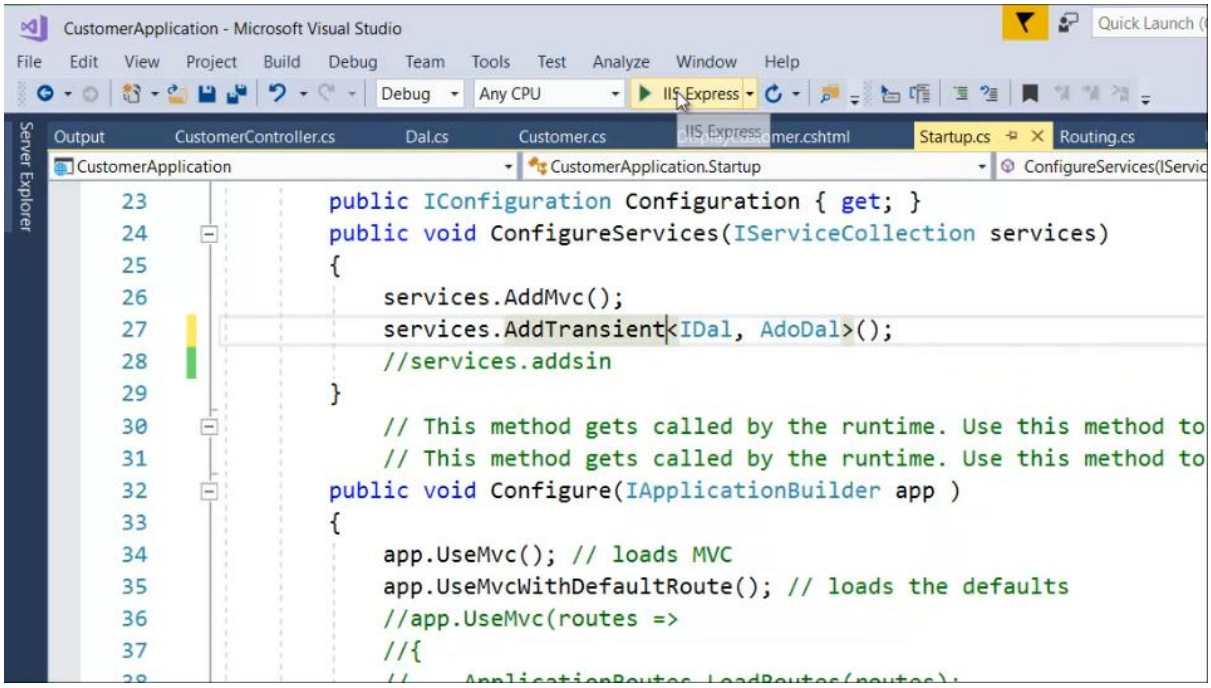
Now lets add property here and see, object that created how many time. Single time or multiple time.

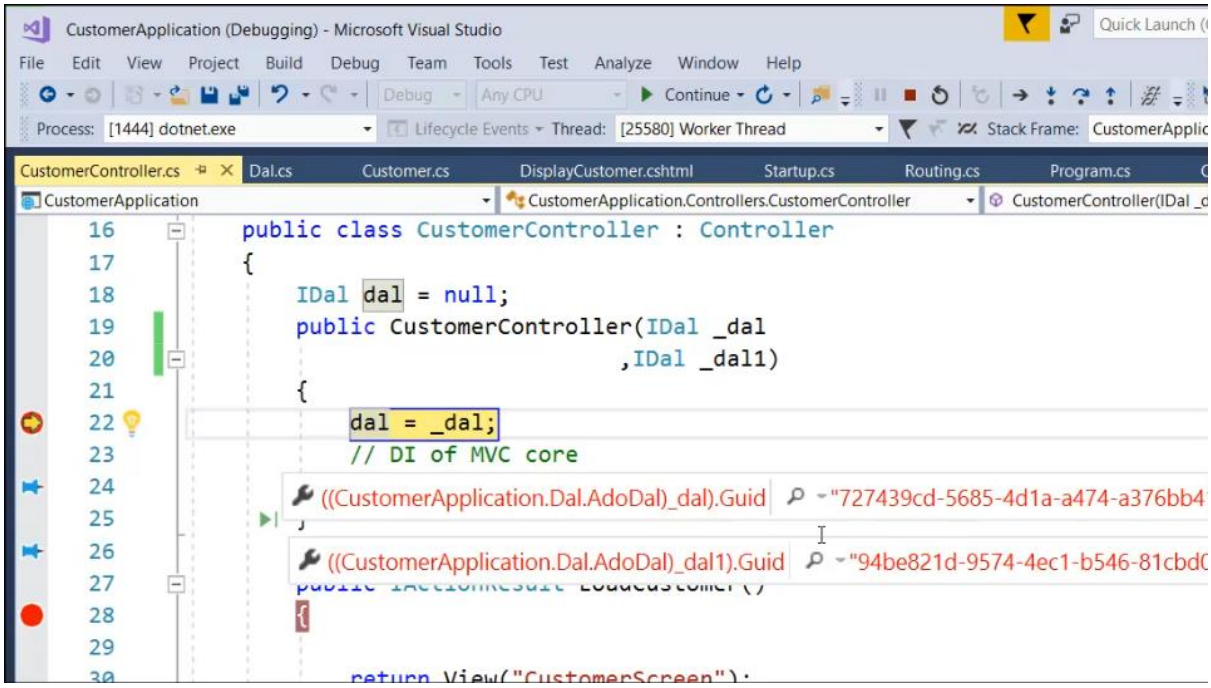


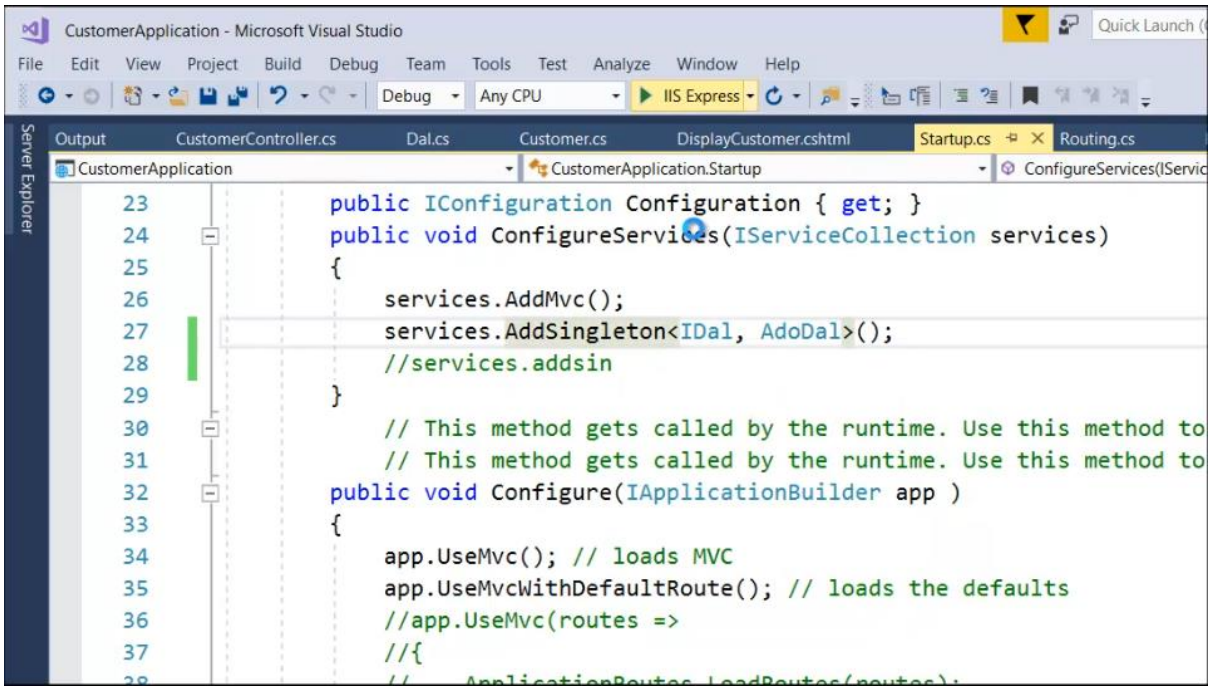

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Output CustomerController.cs Dal.cs Customer.cs DisplayCustomer.cshtml Startup.cs* Routing.cs
CustomerApplication CustomerApplication.Dal.AdoDal AdoDal()
25 }
26 }
27 public class AdoDal : IDal
28 {
29 public string Guid { get; set; }
30 public AdoDal()
31 {
32 this.Guid = System.Guid.NewGuid().ToString();
33 }
34 public void Add()
35 {
36 throw new NotImplementedException();
37 }
38
39 public void Update()

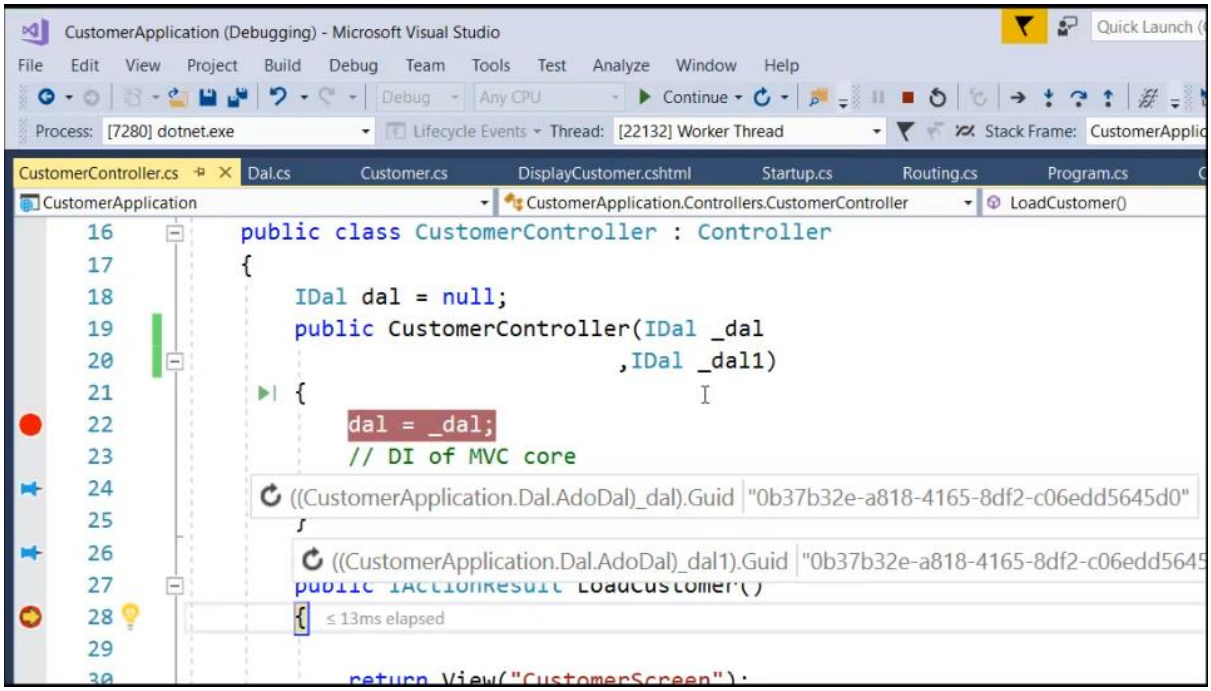

CustomerApplication (Debugging) - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Continue
Process: [18144] dotnet.exe Lifecycle Events Thread: [14668] Worker Thread Stack Frame:
CustomerController.cs Dal.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs Program.cs
CustomerApplication CustomerApplication.Controllers.CustomerController CustomerController(IDal _d
16 public class CustomerController : Controller
17 {
18 IDal dal = null;
19 public CustomerController(IDal _dal
20 ,IDal _dal
_dal {CustomerApplication.Dal.AdoDal}
Guid "8dd051aa-c0fe-4f3a-be4b-920478eb4596"
21 {
22 dal = _dal;
23 // DI of MVC core
24 // inversion of control
25 }
26 // load the customer UI interface
27 public IActionResult LoadCustomer()
28 {
29
30 return View("CustomerScreen");


CustomerApplication (Debugging) - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Continue
Process: [18144] dotnet.exe Lifecycle Events Thread: [14668] Worker Thread Stack Frame:
CustomerController.cs Dal.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs Program.cs
CustomerApplication CustomerApplication.Controllers.CustomerController CustomerController(IDal _d
16 public class CustomerController : Controller
17 {
18 IDal dal = null;
19 public CustomerController(IDal _dal
20 ,IDal _dal1)
21 {
22 dal = _dal;
23 // DI of MVC core
24
25
26
27
28 {
29
30 return View("CustomerScreen");
((CustomerApplication.Dal.AdoDal)_dal).Guid "8dd051aa-c0fe-4f3a-be4b-920478eb
((CustomerApplication.Dal.AdoDal)_dal1).Guid "8dd051aa-c0fe-4f3a-be4b-920478e

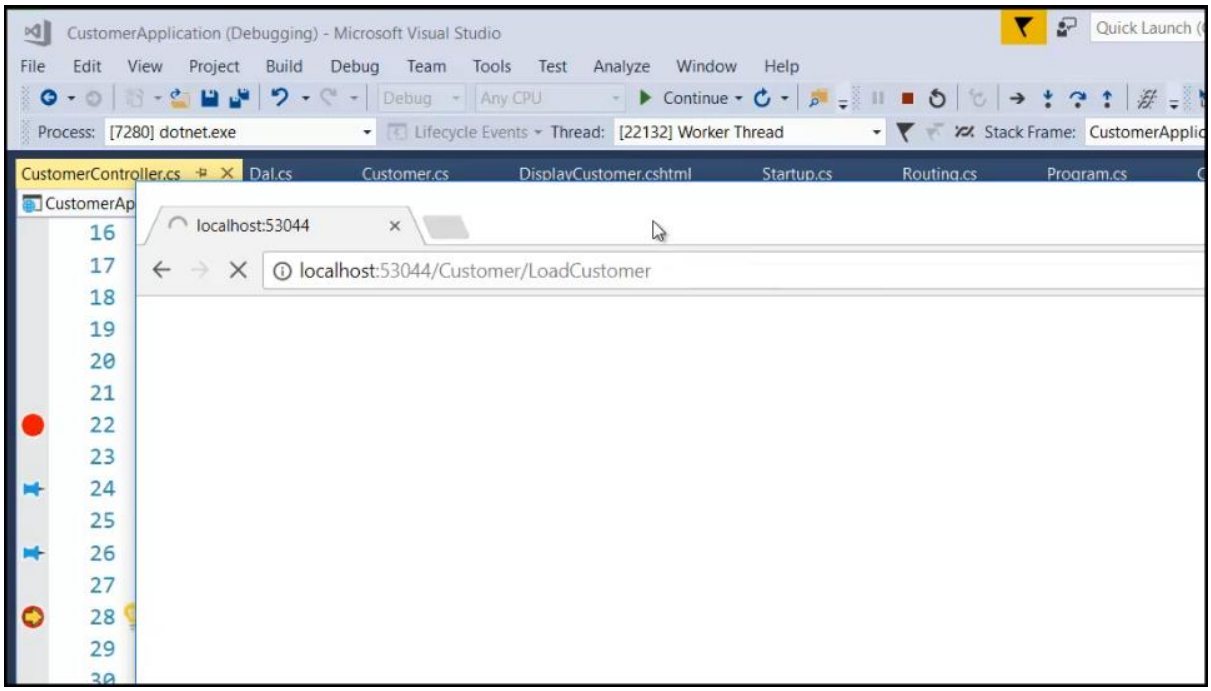

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
IIS Express
Server Explorer
Output CustomerController.cs Dal.cs Customer.cs Startup.cs Routing.cs
CustomerApplication CustomerApplication.Startup ConfigureServices(IServic
23 public IConfiguration Configuration { get; }
24 public void ConfigureServices(IServiceCollection services)
25 {
26 services.AddMvc();
27 services.AddTransient<IDal, AdoDal>();
28 //services.addsin
29 }
30 // This method gets called by the runtime. Use this method to
31 // This method gets called by the runtime. Use this method to
32 public void Configure(IApplicationBuilder app)
33 {
34 app.UseMvc(); // loads MVC
35 app.UseMvcWithDefaultRoute(); // loads the defaults
36 //app.UseMvc(routes =>
37 //{

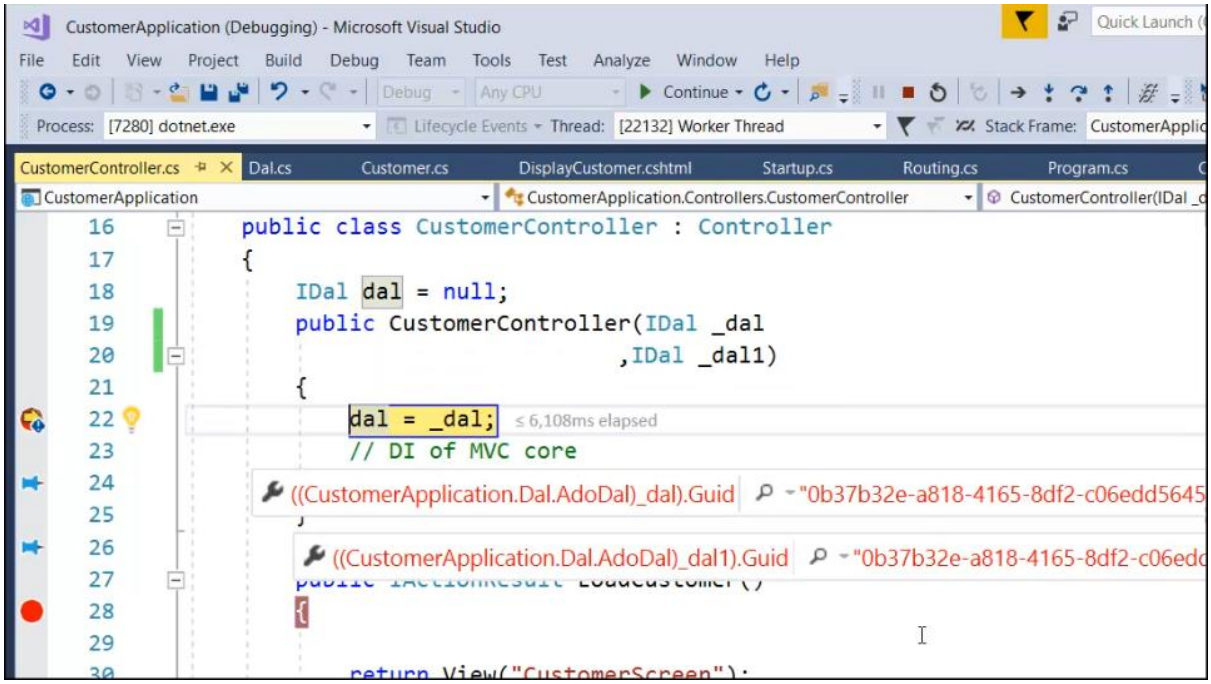

CustomerApplication (Debugging) - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Continue
Process: [1444] dotnet.exe Thread: [25580] Worker Thread
CustomerController.cs Dal.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs Program.cs
CustomerApplication CustomerApplication.Controllers.CustomerController CustomerController(IDal _d
public class CustomerController : Controller
{
IDal dal = null;
public CustomerController(IDal _dal
,IDal _dal1)
{
dal = _dal;
// DI of MVC core
((CustomerApplication.Dal.AdoDal)_dal).Guid "727439cd-5685-4d1a-a474-a376bb4
((CustomerApplication.Dal.AdoDal)_dal1).Guid "94be821d-9574-4ec1-b546-81cbd0
return View("CustomerScreen");


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
Server Explorer
Output CustomerController.cs Dal.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs
CustomerApplication CustomerApplication.Startup ConfigureServices(IServic
public IConfiguration Configuration { get; }
public void ConfigureServices(IServiceCollection services)
{
services.AddMvc();
services.AddSingleton<IDal, AdoDal>();
//services.addsin
}
// This method gets called by the runtime. Use this method to
// This method gets called by the runtime. Use this method to
public void Configure(IApplicationBuilder app)
{
app.UseMvc(); // loads MVC
app.UseMvcWithDefaultRoute(); // loads the defaults
//app.UseMvc(routes =>
//{

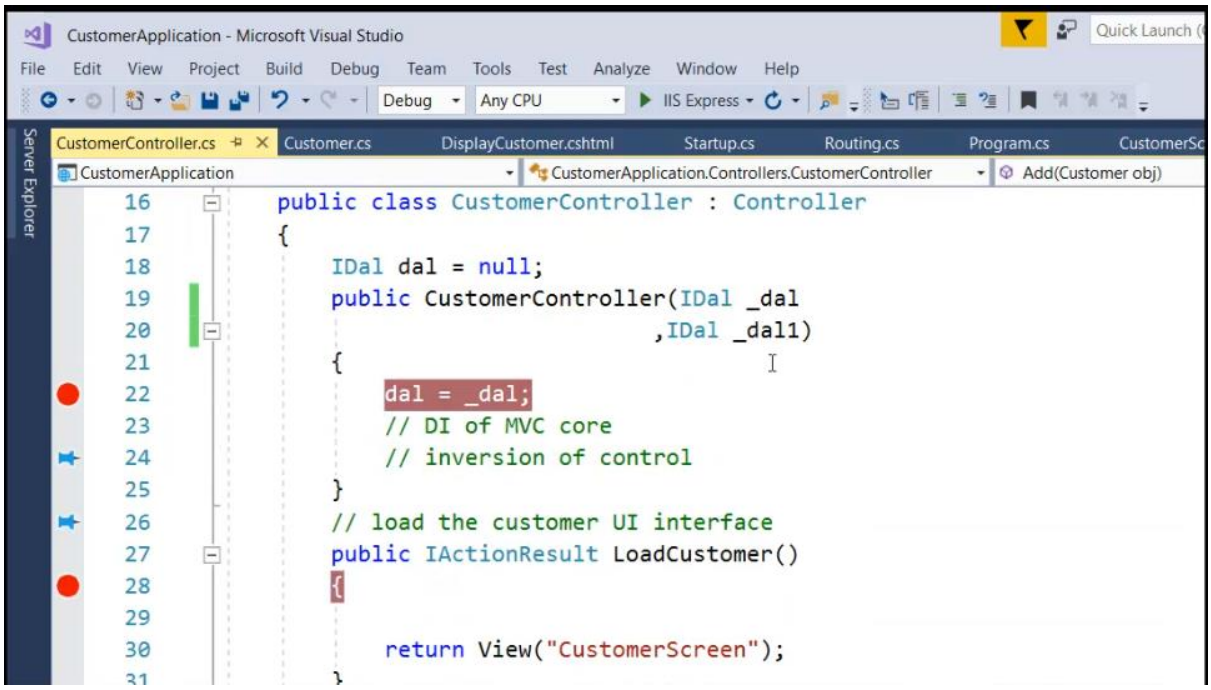


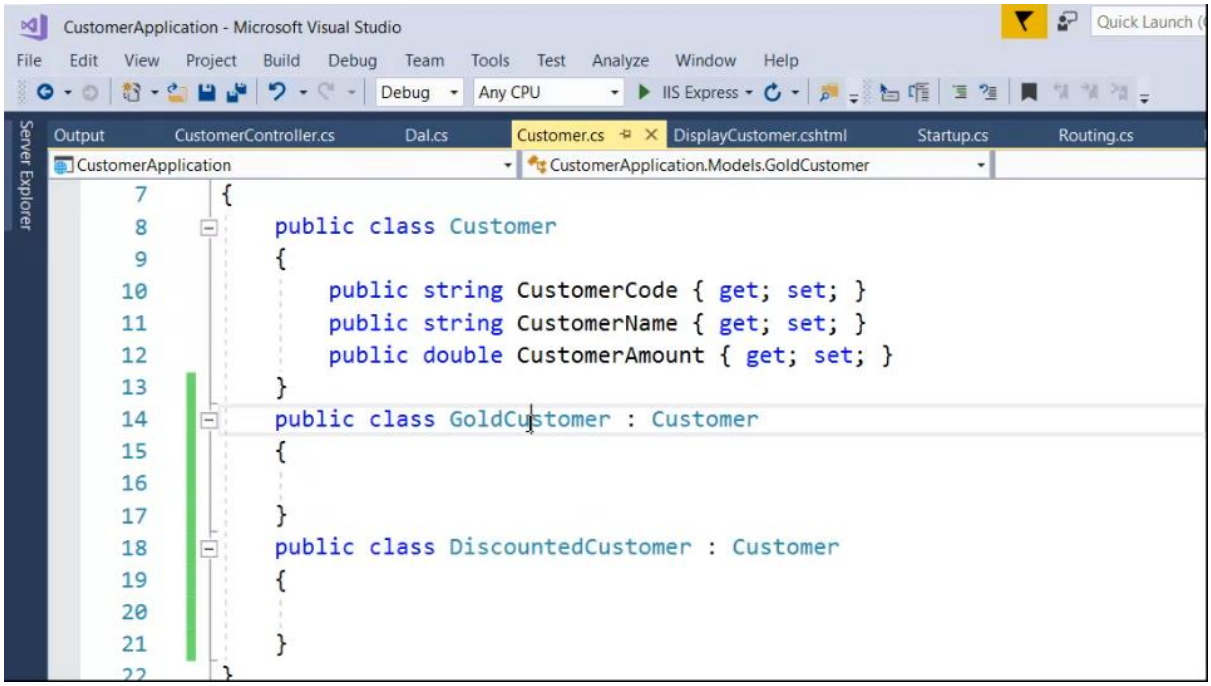
Note : Now hit again the same request again to see.

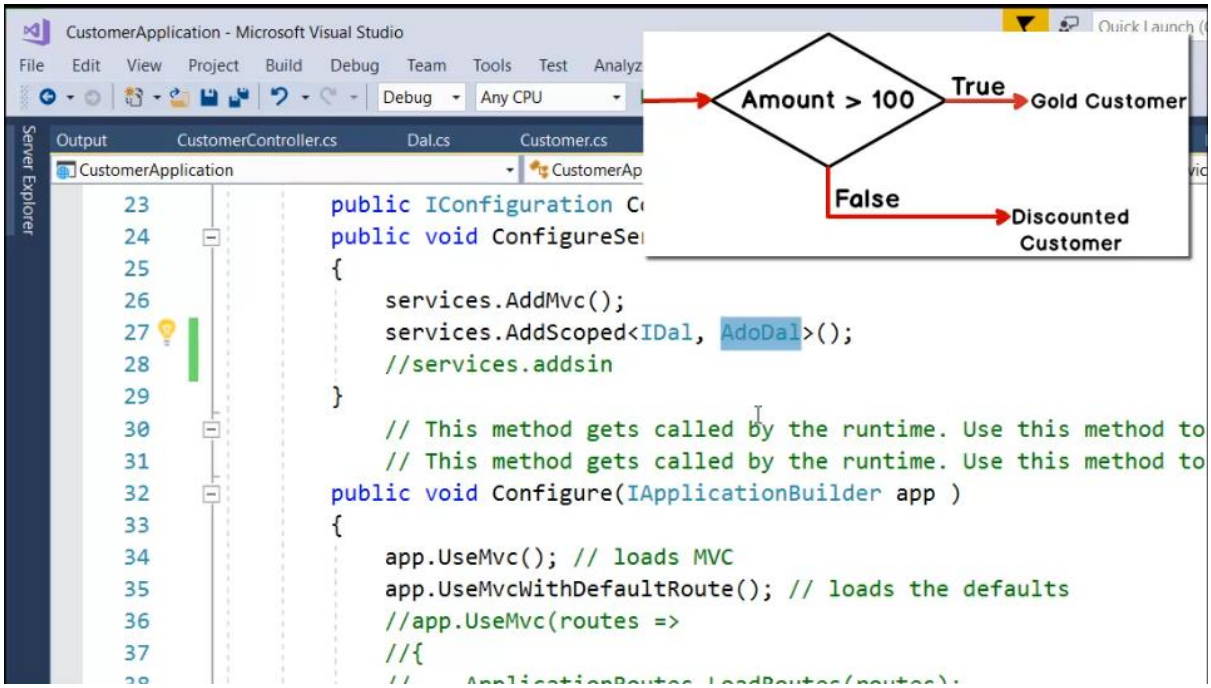


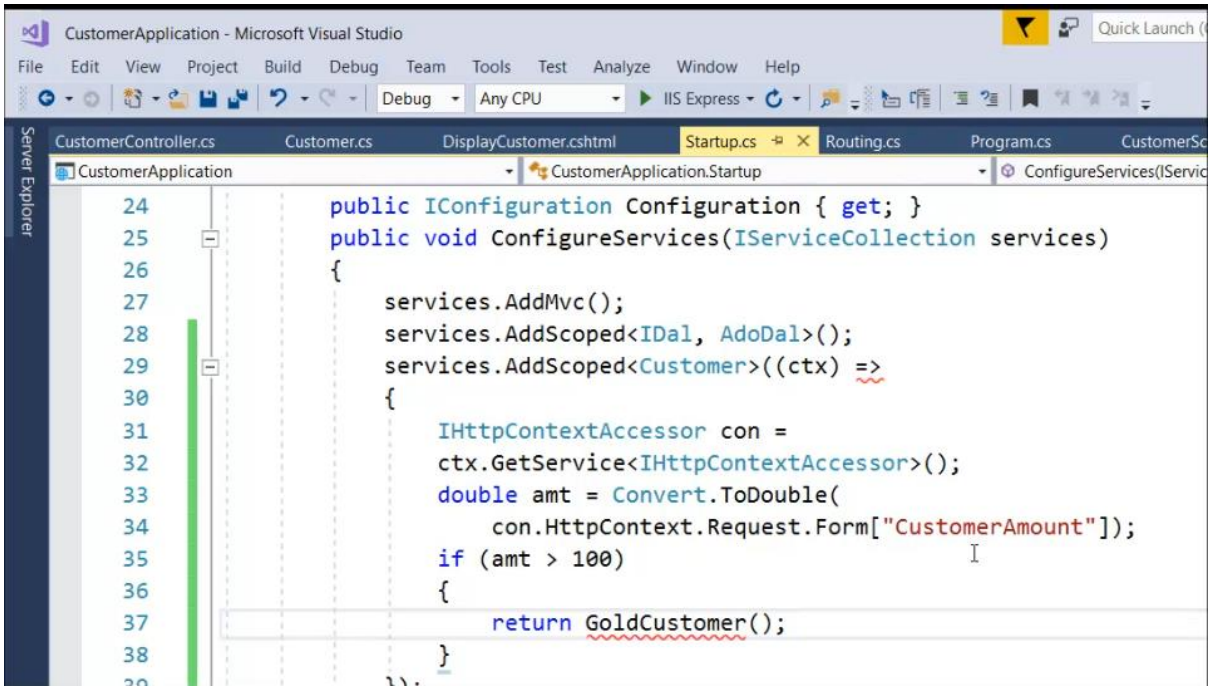


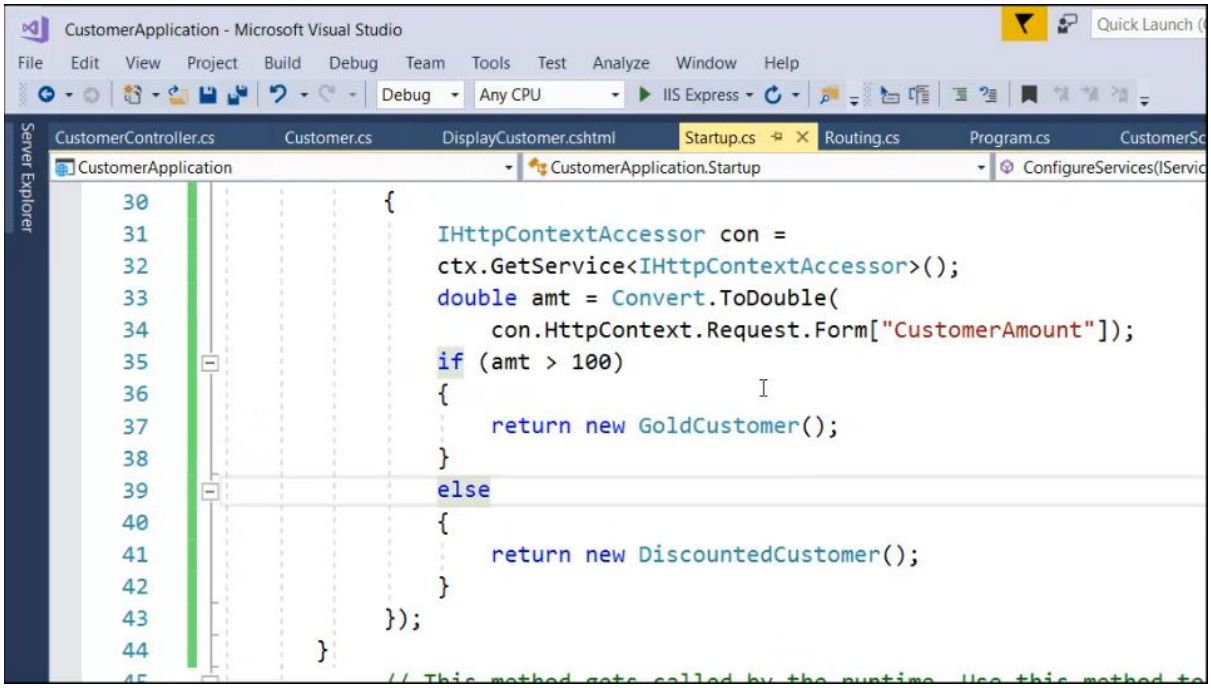
Note : The two ways you create the instance , by using constructor and another way [FromServices]

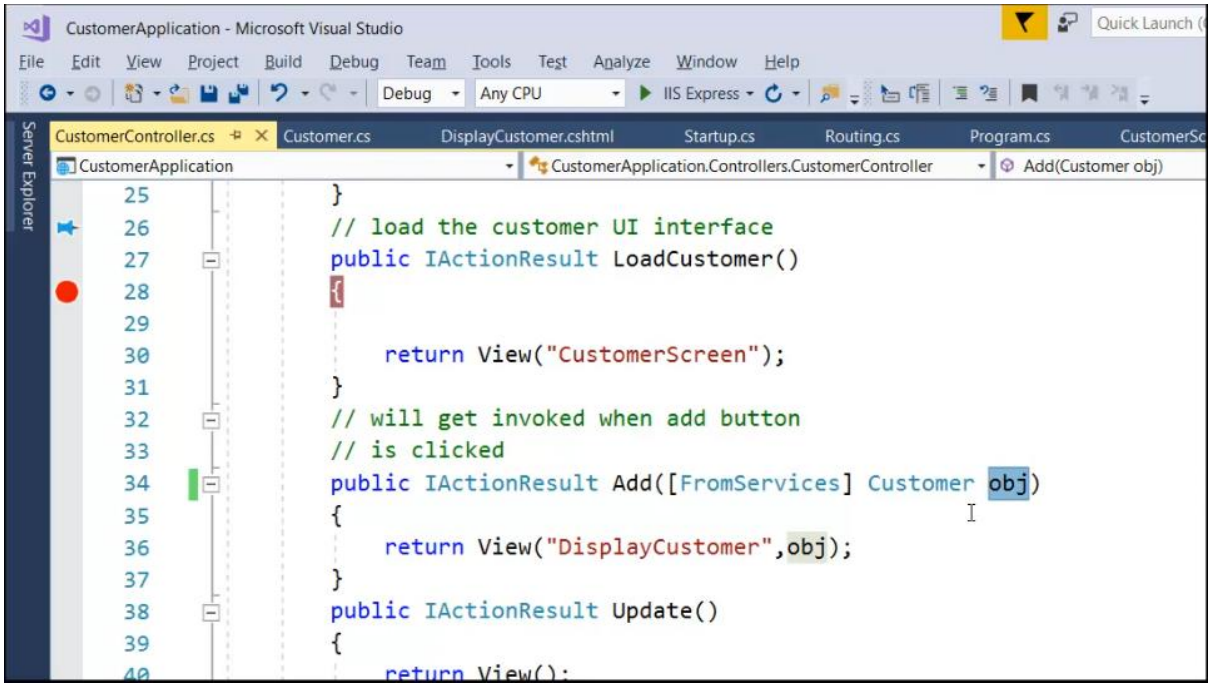



CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
IIS Express
Output
CustomerController.cs
Dal.cs
Customer.cs
DisplayCustomer.cshtml
Startup.cs
Routing.cs
CustomerApplication
CustomerApplication.Models.GoldCustomer
Server Explorer
7 {
8 public class Customer
9 {
10 public string CustomerCode { get; set; }
11 public string CustomerName { get; set; }
12 public double CustomerAmount { get; set; }
13 }
14 public class GoldCustomer : Customer
15 {
16
17 }
18 public class DiscountedCustomer : Customer
19 {
20
21 }
22 }

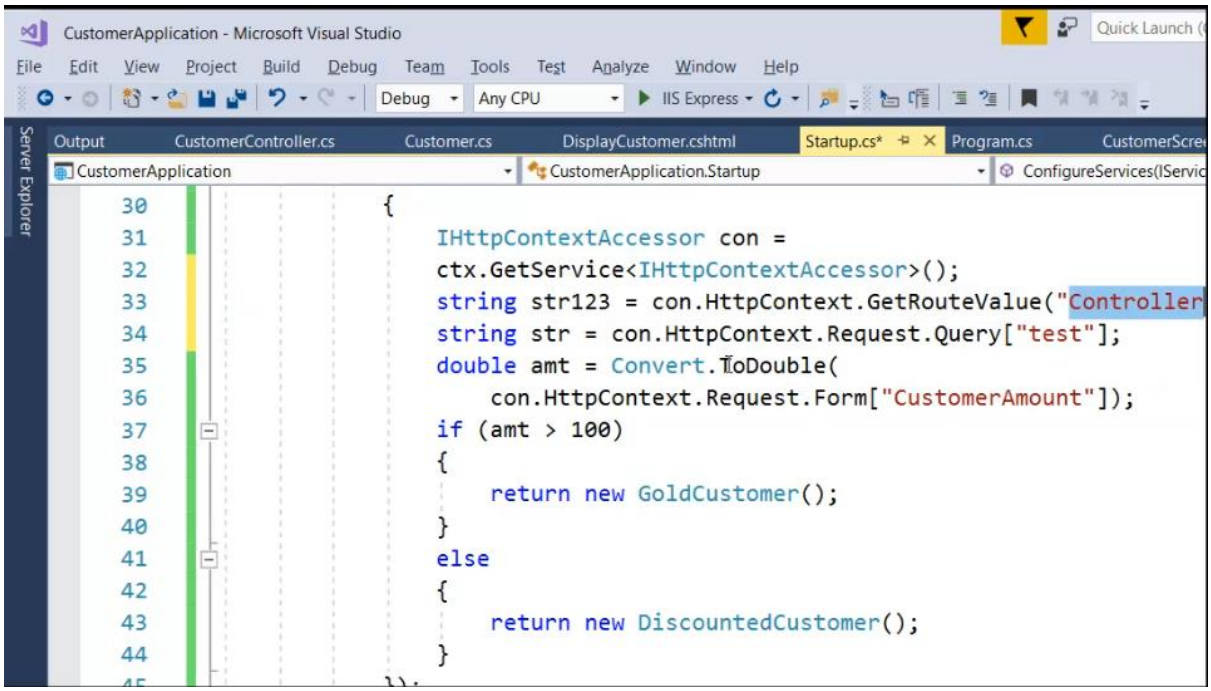

CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyz
Debug
Any CPU
Output
CustomerController.cs
Dal.cs
Customer.cs
CustomerApplication
Server Explorer
Amount > 100
True
Gold Customer
False
Discounted Customer
23 public IConfiguration C
24 public void ConfigureSe
25 {
26 services.AddMvc();
27 services.AddScoped<IDal, AdoDal>();
28 //services.addsin
29 }
30 // This method gets called by the runtime. Use this method to
31 // This method gets called by the runtime. Use this method to
32 public void Configure(IApplicationBuilder app)
33 {
34 app.UseMvc(); // loads MVC
35 app.UseMvcWithDefaultRoute(); // loads the defaults
36 //app.UseMvc(routes =>
37 //{


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
CustomerController.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs Program.cs
CustomerApplication CustomerApplication.Startup ConfigureServices(IServic
Server Explorer
24 public IConfiguration Configuration { get; }
25 public void ConfigureServices(IServiceCollection services)
26 {
27 services.AddMvc();
28 services.AddScoped<IDal, AdoDal>();
29 services.AddScoped<Customer>((ctx) =>
30 {
31 IHttpContextAccessor con =
32 ctx.GetService<IHttpContextAccessor>();
33 double amt = Convert.ToDouble(
34 con.HttpContext.Request.Form["CustomerAmount"]);
35 if (amt > 100)
36 {
37 return GoldCustomer();
38 }


CustomerApplication - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express
CustomerController.cs Customer.cs DisplayCustomer.cshtml Startup.cs Routing.cs Program.cs
CustomerApplication CustomerApplication.Startup ConfigureServices(IServic
Server Explorer
30 {
31 IHttpContextAccessor con =
32 ctx.GetService<IHttpContextAccessor>();
33 double amt = Convert.ToDouble(
34 con.HttpContext.Request.Form["CustomerAmount"]);
35 if (amt > 100)
36 {
37 return new GoldCustomer();
38 }
39 else
40 {
41 return new DiscountedCustomer();
42 }
43 });
44 }



By Condition , the FromService customer obj , its depends on factory code we have written into this start-up file.



What is IOC and DI conceptually?
In Which file do we define MVC CORE DI objects?
Differentiate between Transient, Scoped, Singleton and Factory.
What is the syntax of DI in MVC core?
What is the use of ContextAccessor and how do we get it ?