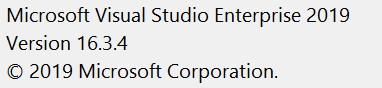
**Create ASP.NET Core API using AWS DynamoDB**

1. Tools required for this project:
2. Visual Studio (I am using this:



1. AWS CLI

Msi installer for windows (just install and open cmd)

<https://docs.aws.amazon.com/cli/latest/userguide/install-windows.html#install-msi-on-windows>

 Download and Install aws-shell.

Just install and open cmd, and run aws configure command

(use access key id and secret access key that you saved earlier)

C: configure 
Access Key ID 
Secret Access Key 
IY+gTocr55LeqyK9RAiLJeu•FNb37voqMNdJb99c7 
Default region us-west-2 
Default Output format (None) : 
C \ users\Admi nistrator> 

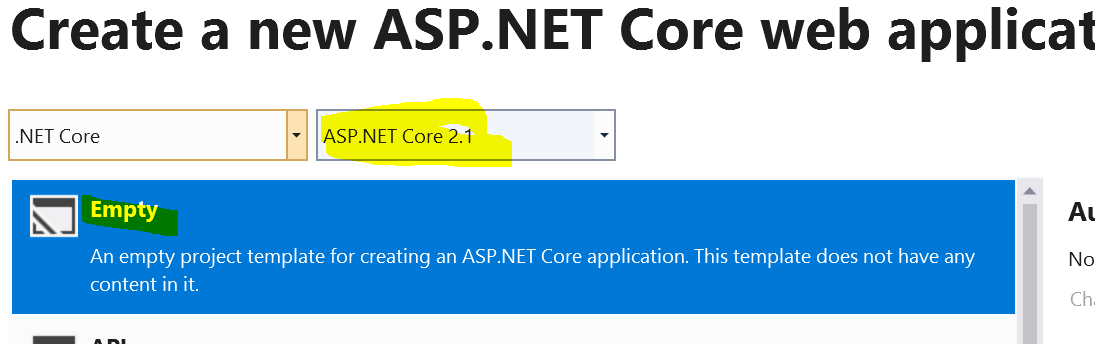
(Now you will see .aws files in C:\Users\{current user name}\.aws)

The database client uses this id and key to access database in cloud.

1. Postman

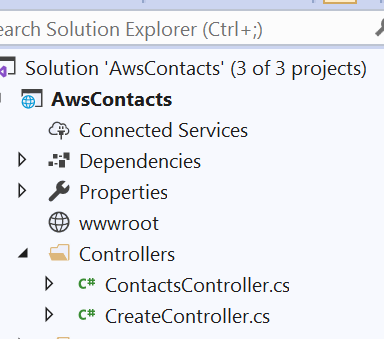
<https://www.getpostman.com/downloads/>

1. Now you are all set to create an ASP.NET API project
2. Choose ASP.NET Core Web application project. Click Next
3. Name the project AwsContacts
4. Make sure it is empty project and its version is 2.1 (version conflict causes problem)



1. Create the project.
2. Add Controllers folder to project and add empty ContactsController

(Also add empty controller called CreateController)



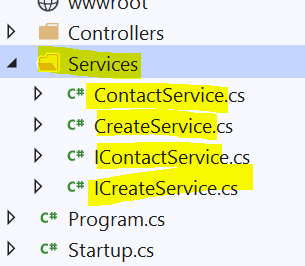
1. Right click AwsContacts project and add new folder (name it Services)

Right click the Services folder and add two interfaces (name them IContactService and ICreateService)

Also add two classes to the folder: ContactService and CreateService

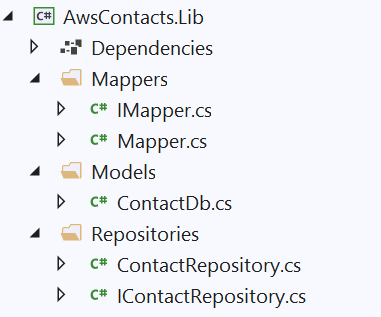
Make sure all interfaces and classes are public

Also make the classes inherit from respective interfaces.



1. Right click the solution and add a project called AwsContacts.Lib

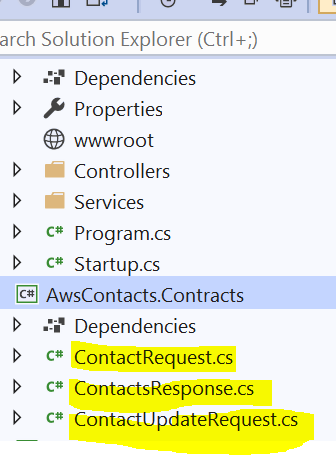
Add these folders and classes to this project:



Make sure all classes and interfaces are public and classes inherit related interfaces.

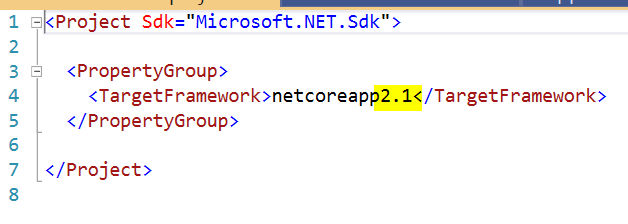
1. Again right click the solution and add another project called AwsContacts.Contracts

Add these three classes to this project



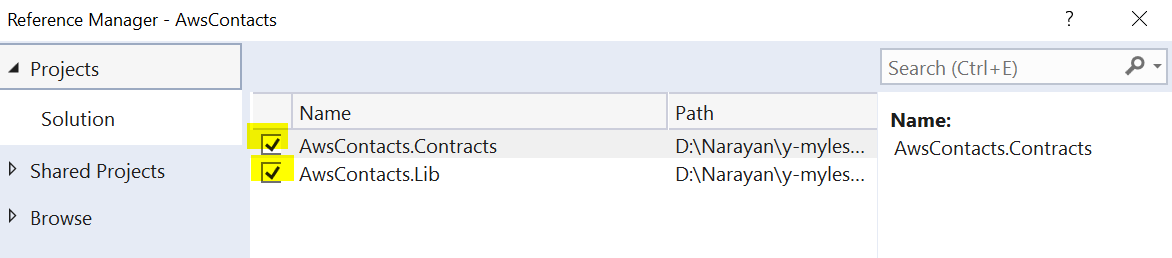
Make sure these classes are public.

1. Now right click AwsContacts.Contract, click Edit Project File and ensure that version is 2.1



If others, just edit this to 2.1.

1. Repeat 7 to another added project as well.
2. Now right click AwsContacts project > Add > Reference and choose the following:



1. Now open your startup.cs file and add this inside ConfigureServices method

services.AddMvc();

//.AddControllersAsServices();

services.AddAWSService<IAmazonDynamoDB>();

services.AddDefaultAWSOptions(new AWSOptions

{

Region = RegionEndpoint.GetBySystemName("us-west-2")

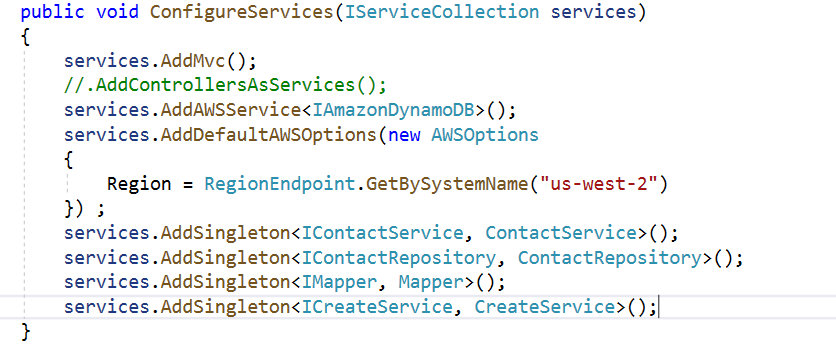
}) ;

services.AddSingleton<IContactService, ContactService>();

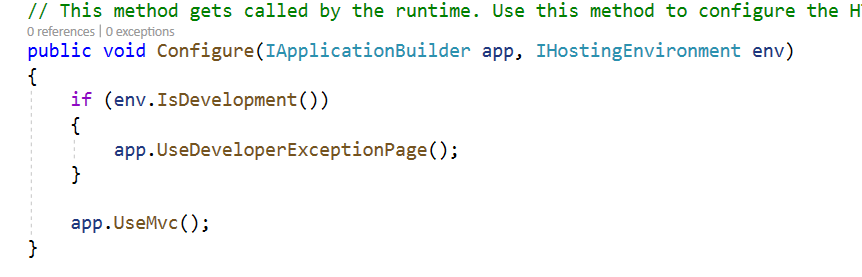
services.AddSingleton<IContactRepository, ContactRepository>();

services.AddSingleton<IMapper, Mapper>();

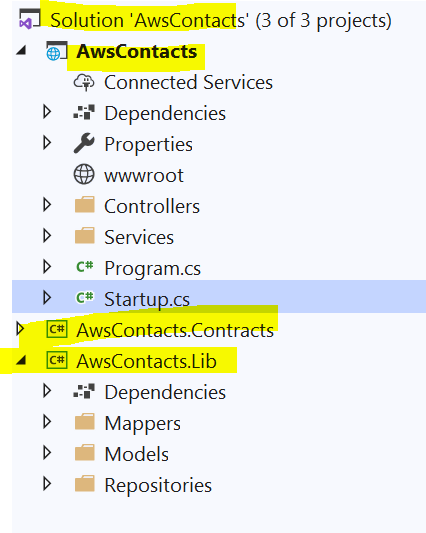
services.AddSingleton<ICreateService, CreateService>();



1. Now make sure your configure method looks like this:



1. The final skeleton of this project will look like this:



1. This project makes better sense if you understand the concepts of middleware and dependency injection in Asp.net core app:
2. Middleware: <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/?view=aspnetcore-3.0>
3. Dependency injection: <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection?view=aspnetcore-3.0>
4. The most common request-response flow in the project is:

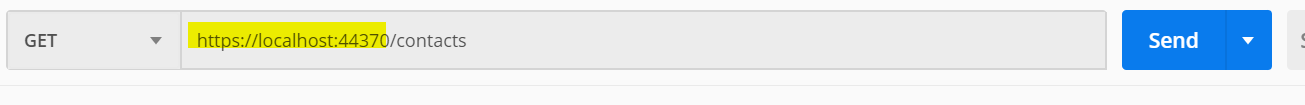
Controller method -> Service class method -> Repository class method -> Mapper class method

1. Now copy each controller method from Github repo and then keep copying corresponding methods in other classes too. That way you will understand the flow.

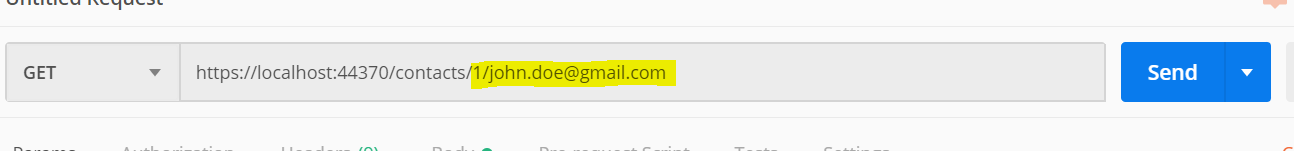
(this forces you to follow request-response flow in 14)

<https://github.com/narayanpg/AwsContacts>

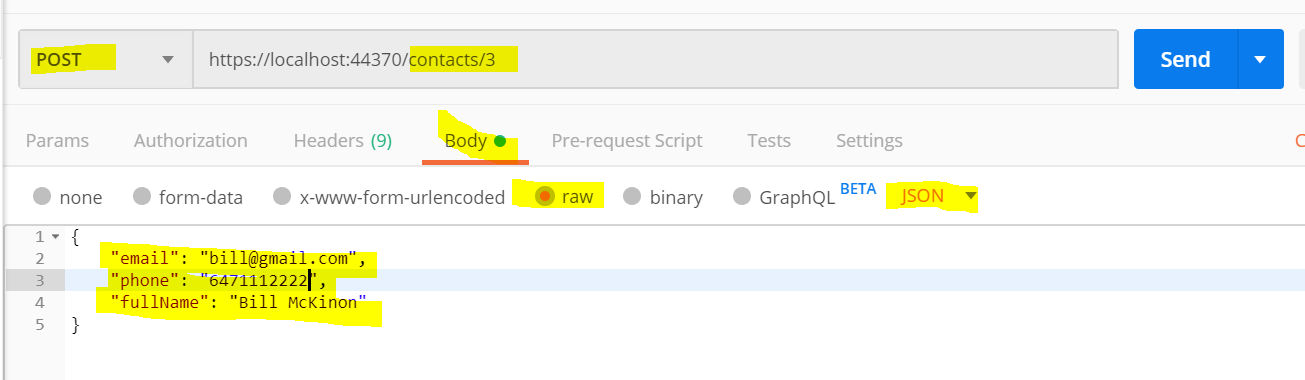
1. After completing each controller methods, test them with Postman.
2. Show all contacts



1. Show contacts by userid and email

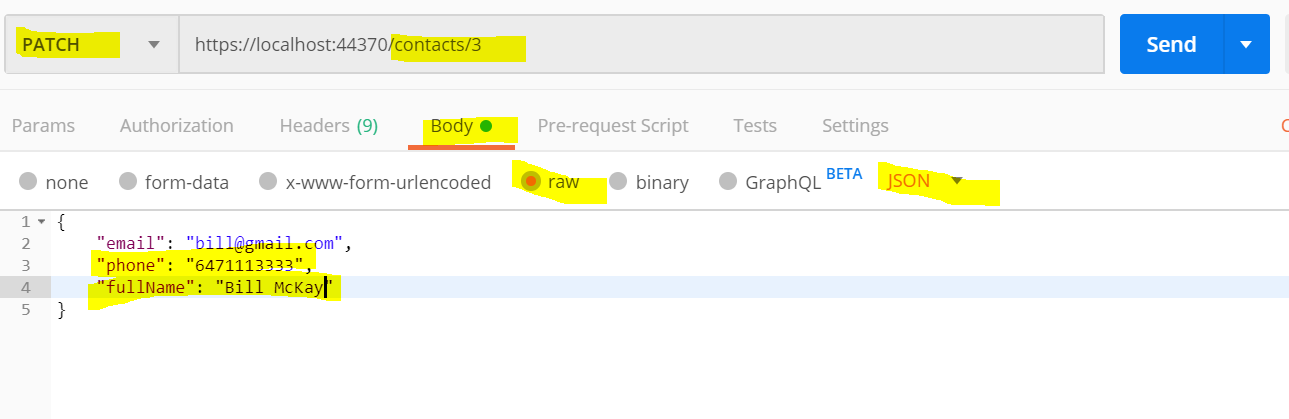


1. Add a contact

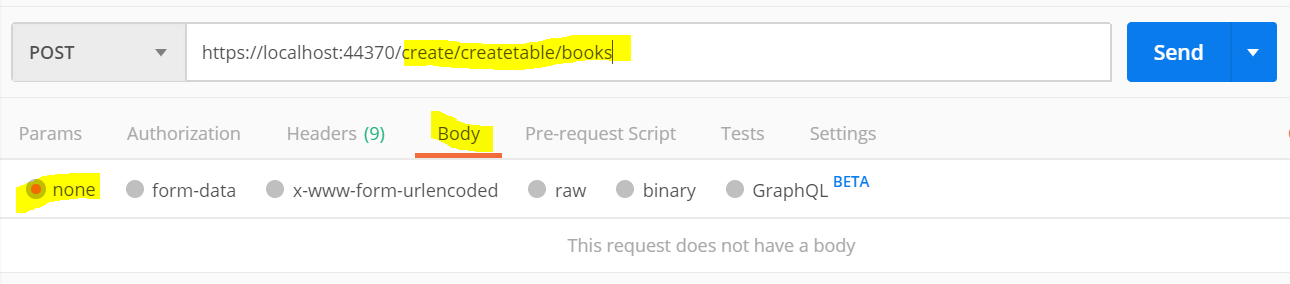


1. Edit a contact

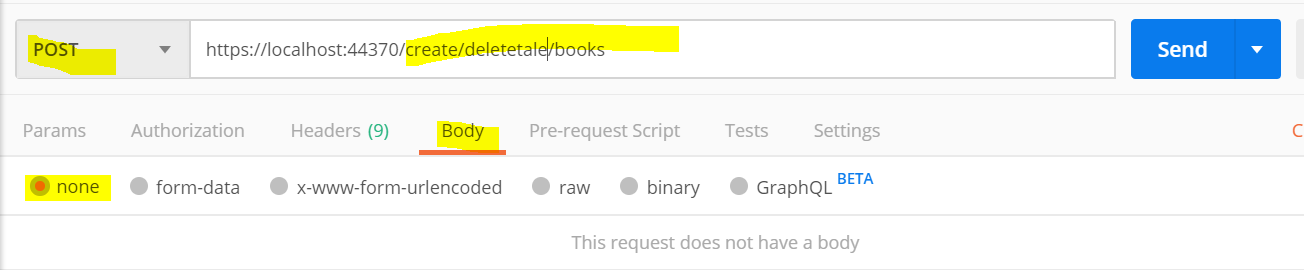
(only phone and fullname can be edited)



1. Create a new table in AWS dynamodb



1. Delete a table in aws dynamoddb



The End