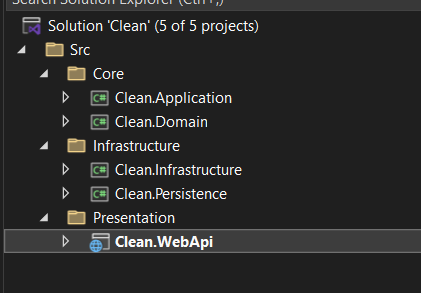
**1) folder setup**

1. create empty asp.net core project
2. solution created now delete project
3. inside it create folders and projects using class library (layers)
4. remove classes eg class1,class2 etc
5. build solution by clicking on build tab
6. set WebAPi as startup project
7. run by play btn



1. **add project refrence & DI**
   1. add refrence by clicking on dependencies

domain: no one

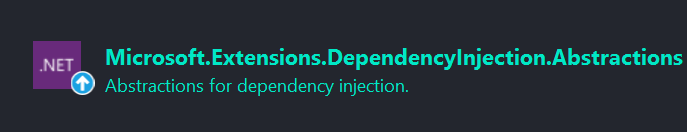
application : domain

infrastructure: application

persistence: application

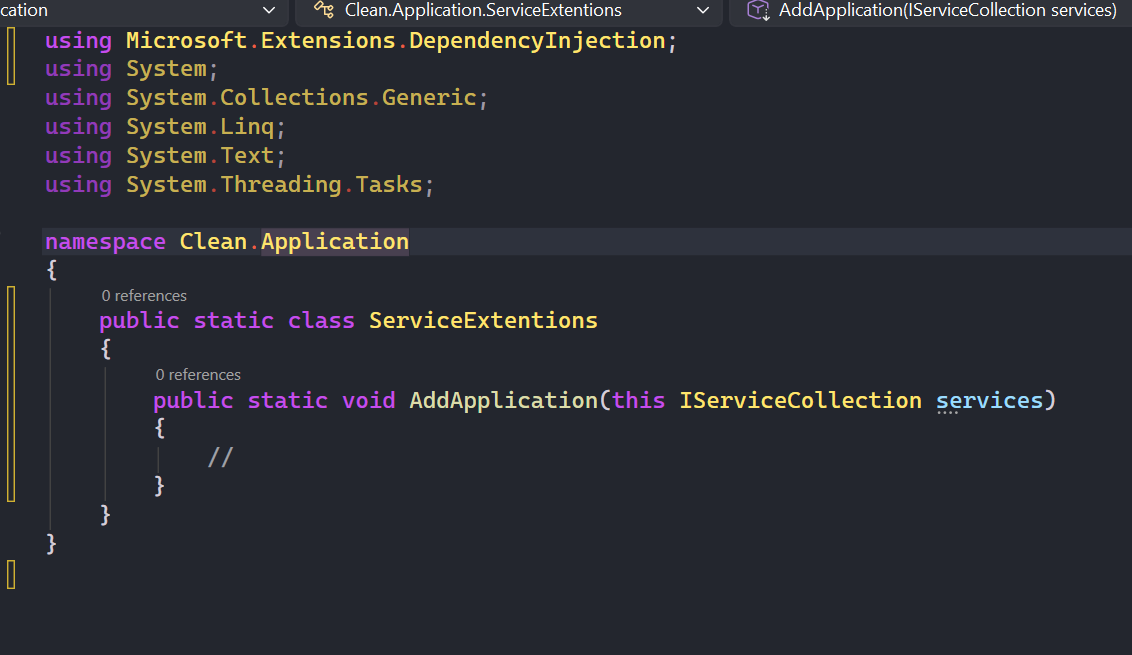
api(presentation):app,per,inf

* 1. add below package in application layer for di (latest)



* 1. create service extension class in Extensions folder app,inf,persis and register in program.cs

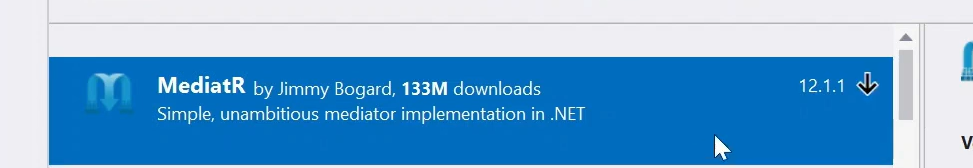




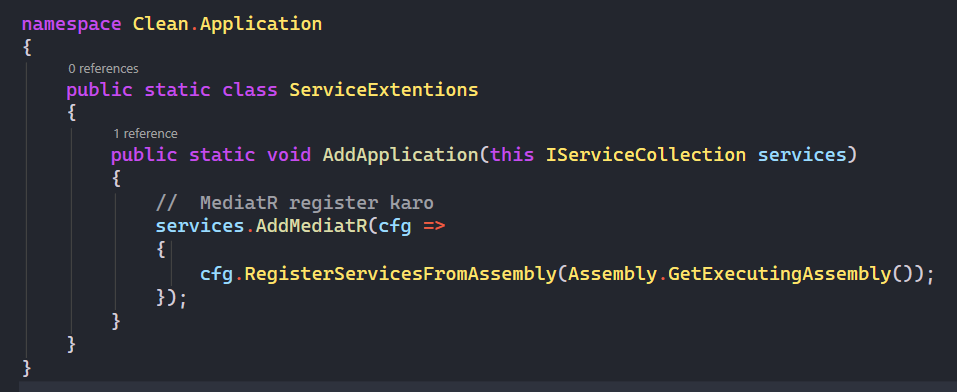


* For all 3

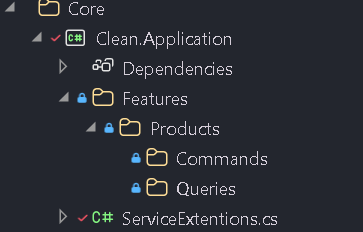
1. Install mediatr in application layer (latest) if old also install di



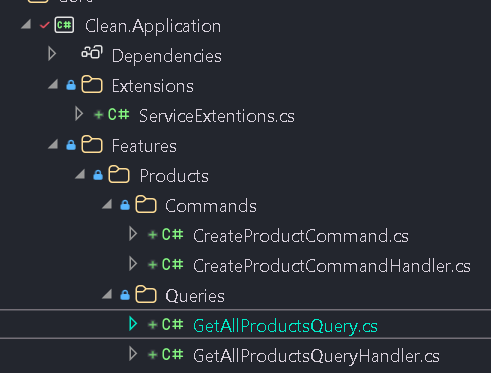
1. Register mediatr in application layer service extension



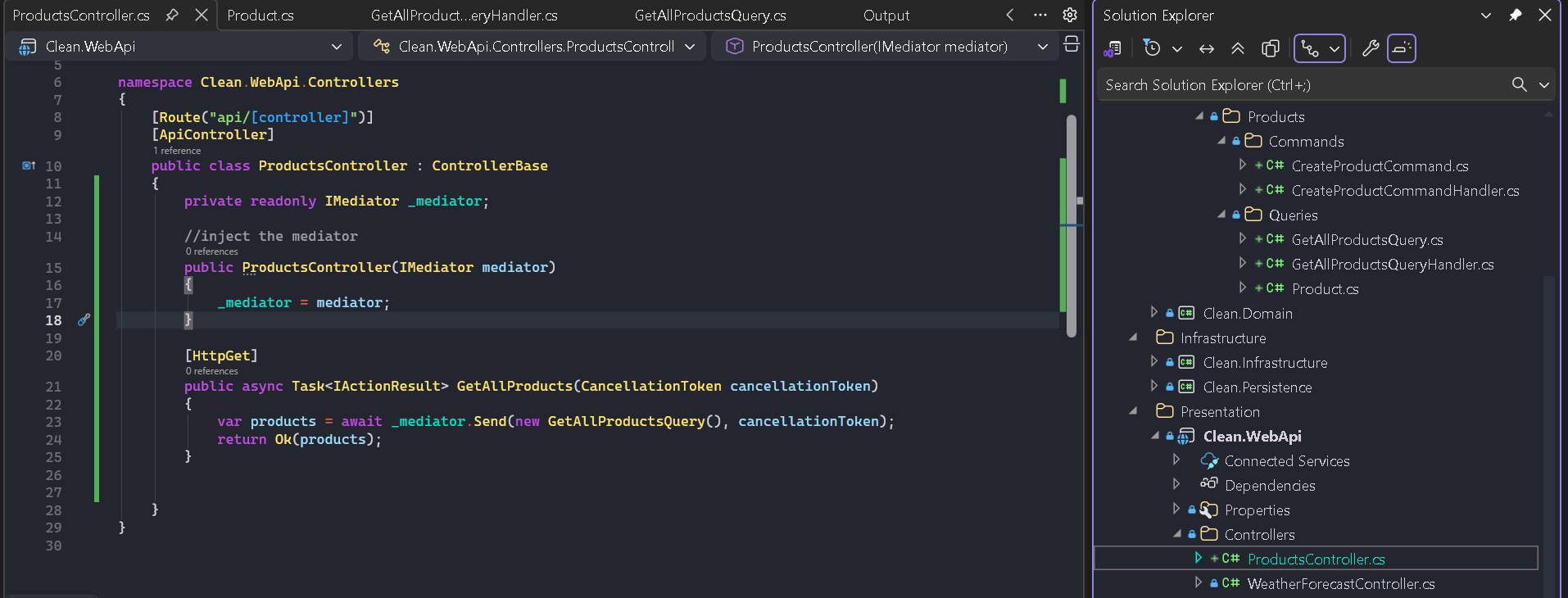
1. Create folder structure note feature can change according to project inside it command and query folder



1. Create command / query for product check **commit 2** for details:



1. Create controller and inject mediatr and create end point check

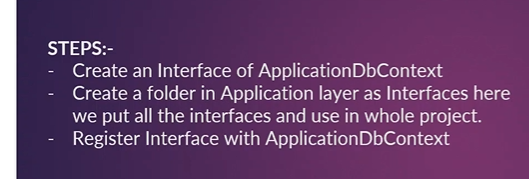


1. Adding ef core and setting up domain layer etc check **commit3** for details



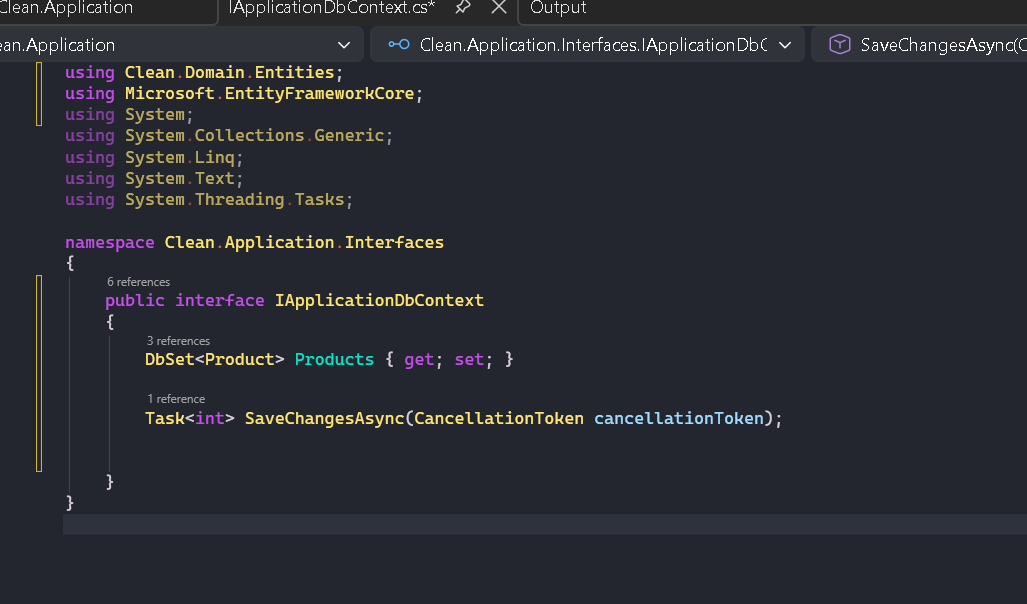
Crud operations

Create interface in application layer



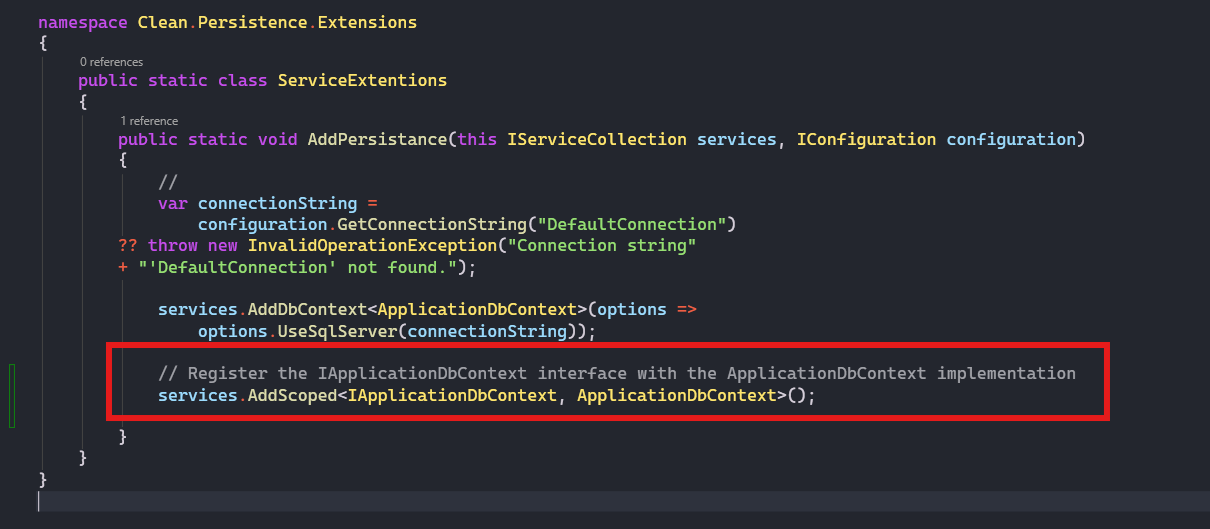
Put all prop / functions in interface

Install ef core package



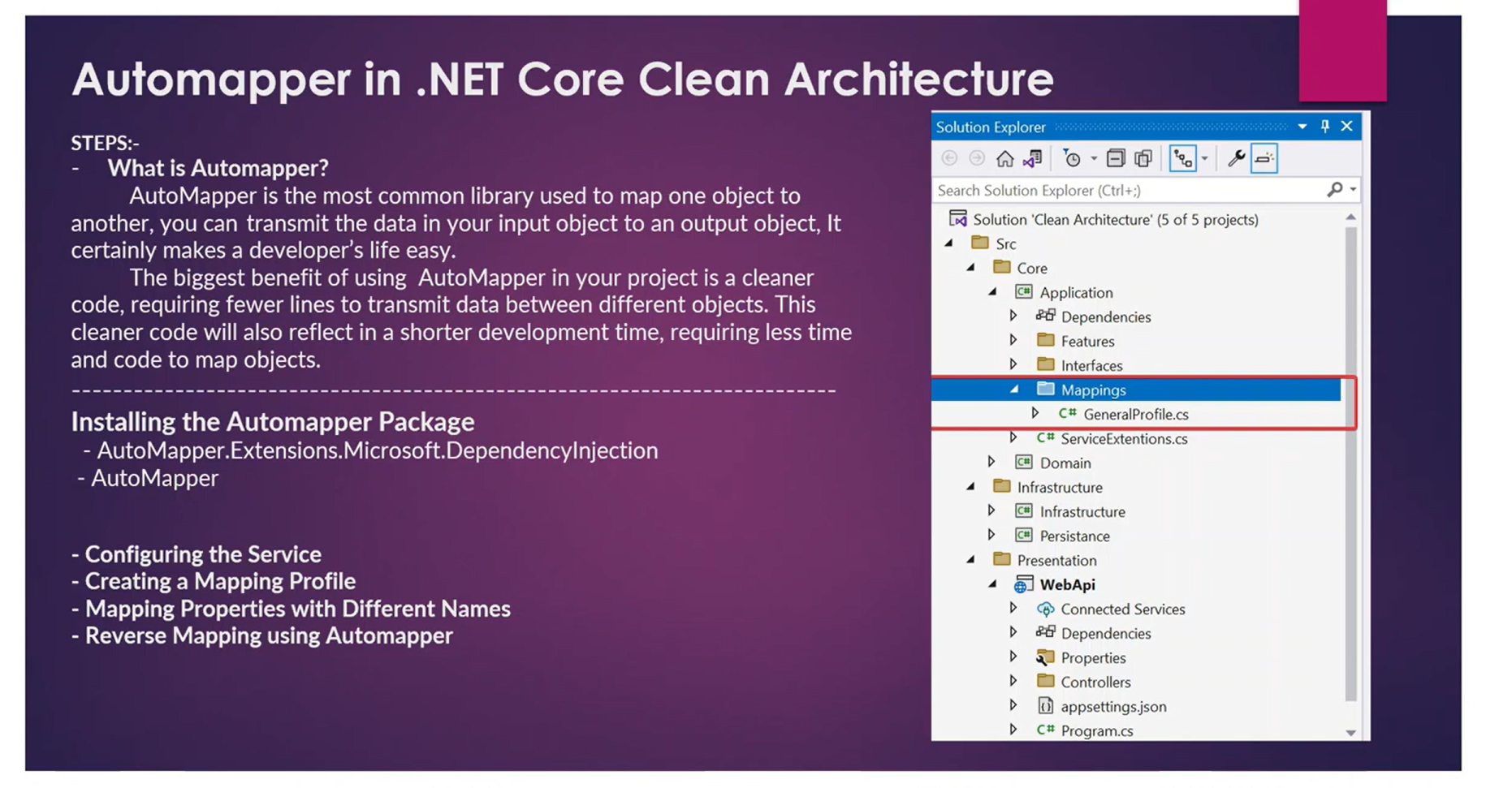
Inherit application db context with interface we created

Register in service extension in persistence



Now crud for this check handlers/commands etc and controller in commit 4

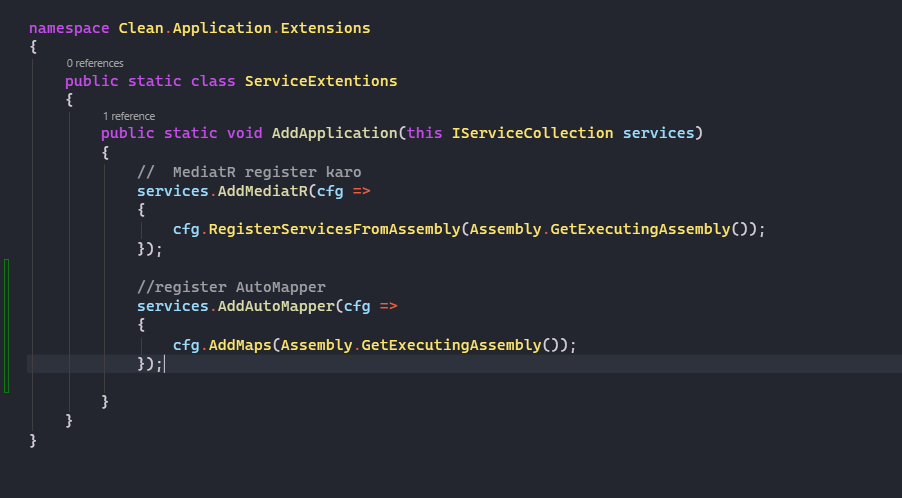
Automapper



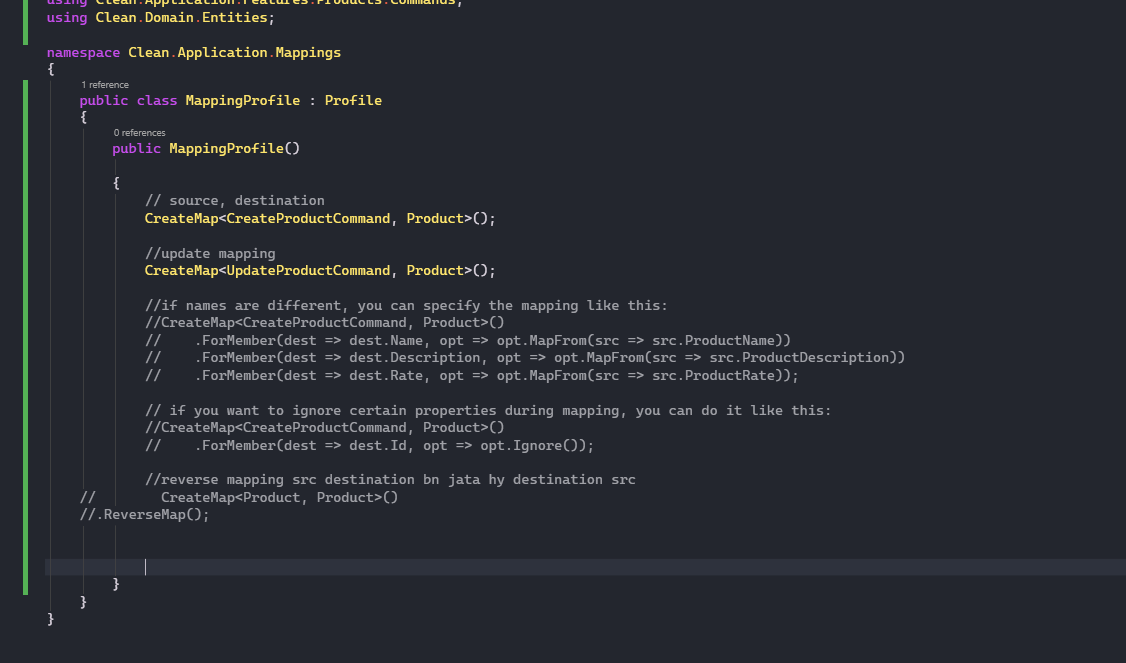


Install package in application layer

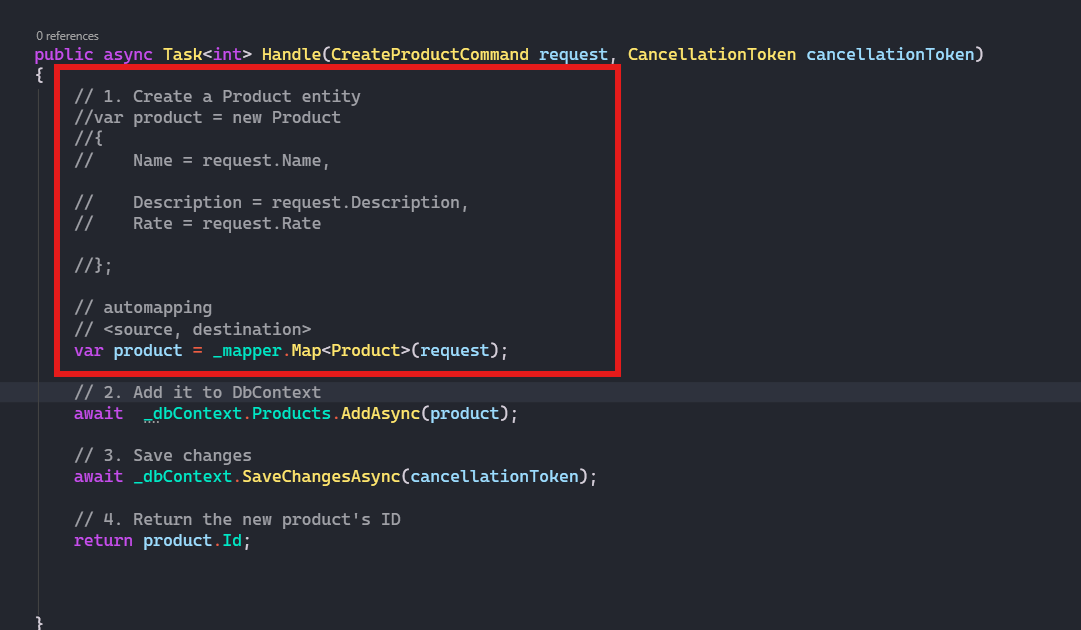
Register in application layer extension method

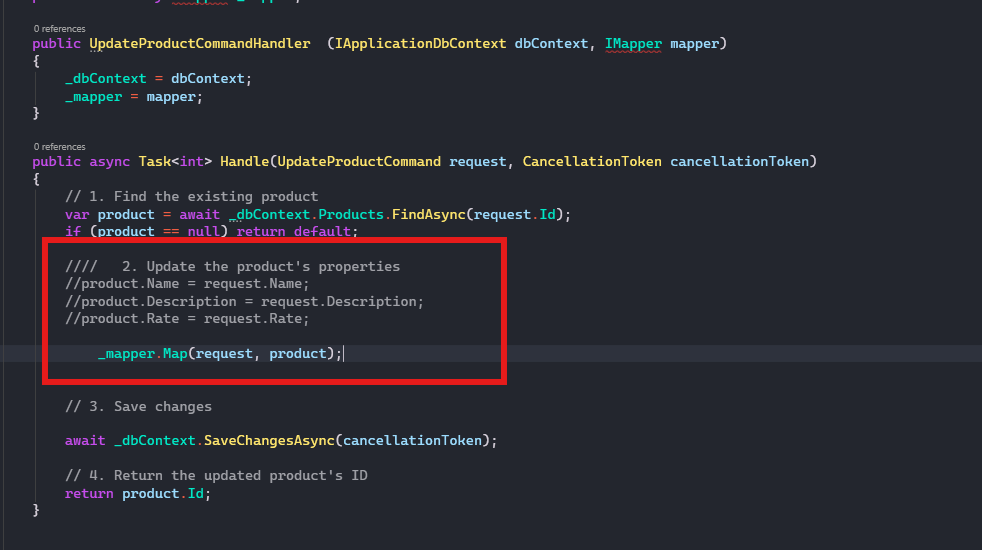


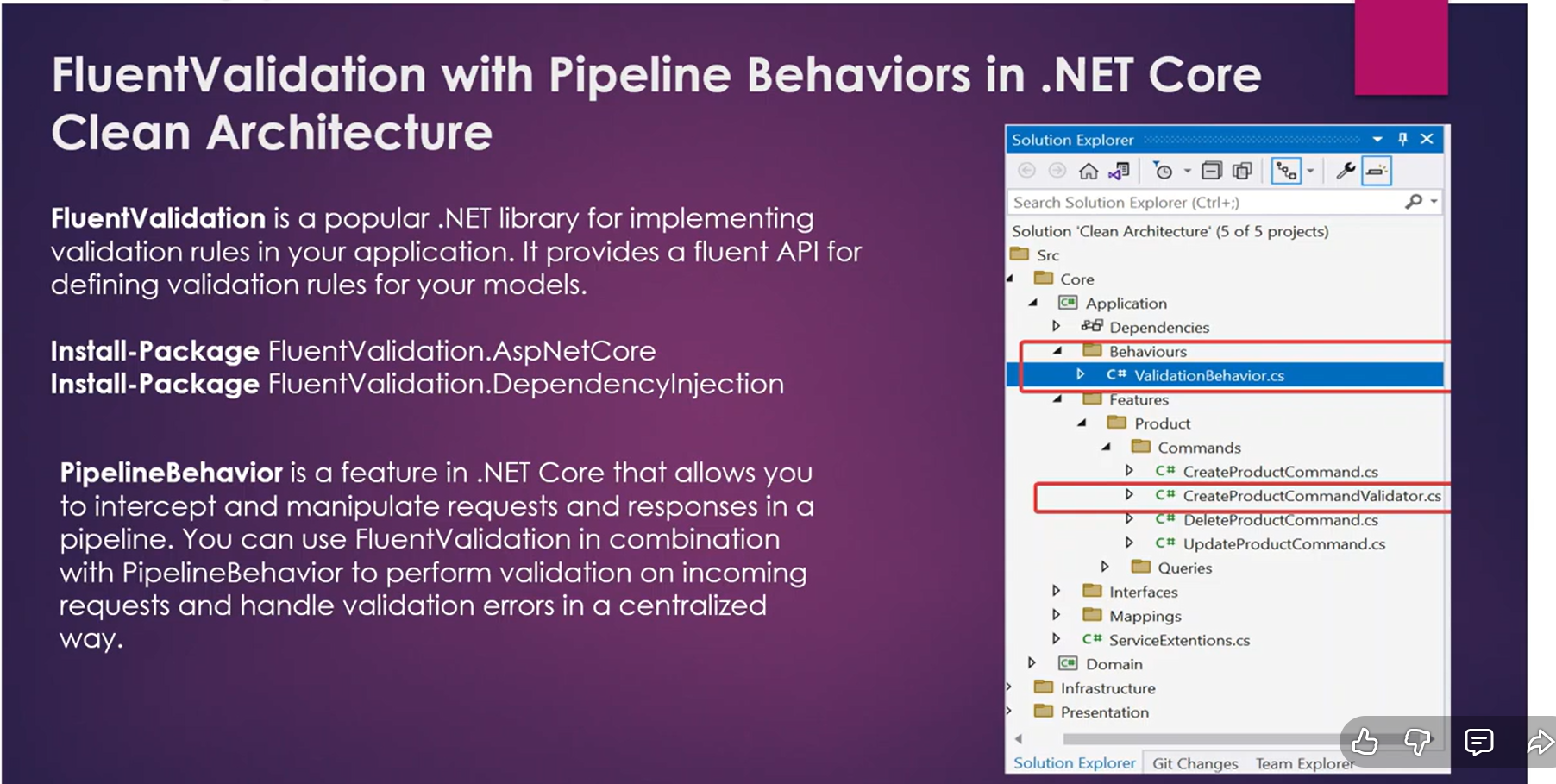
Create mapping profile folder and inside it class in app layer



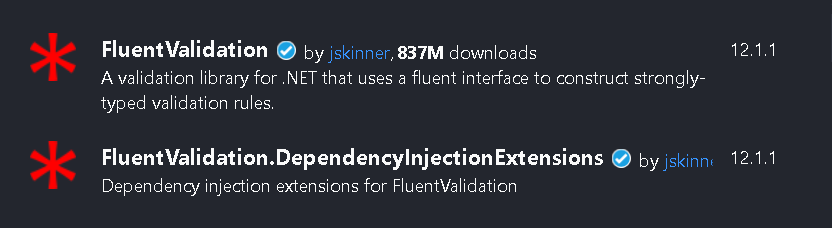
Update the mapping



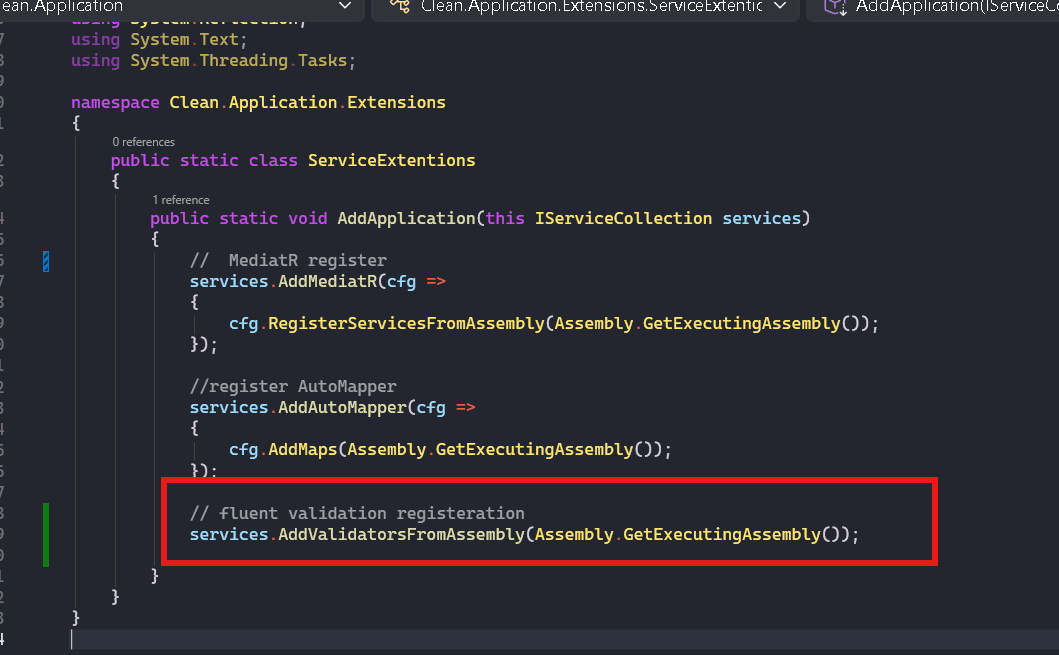




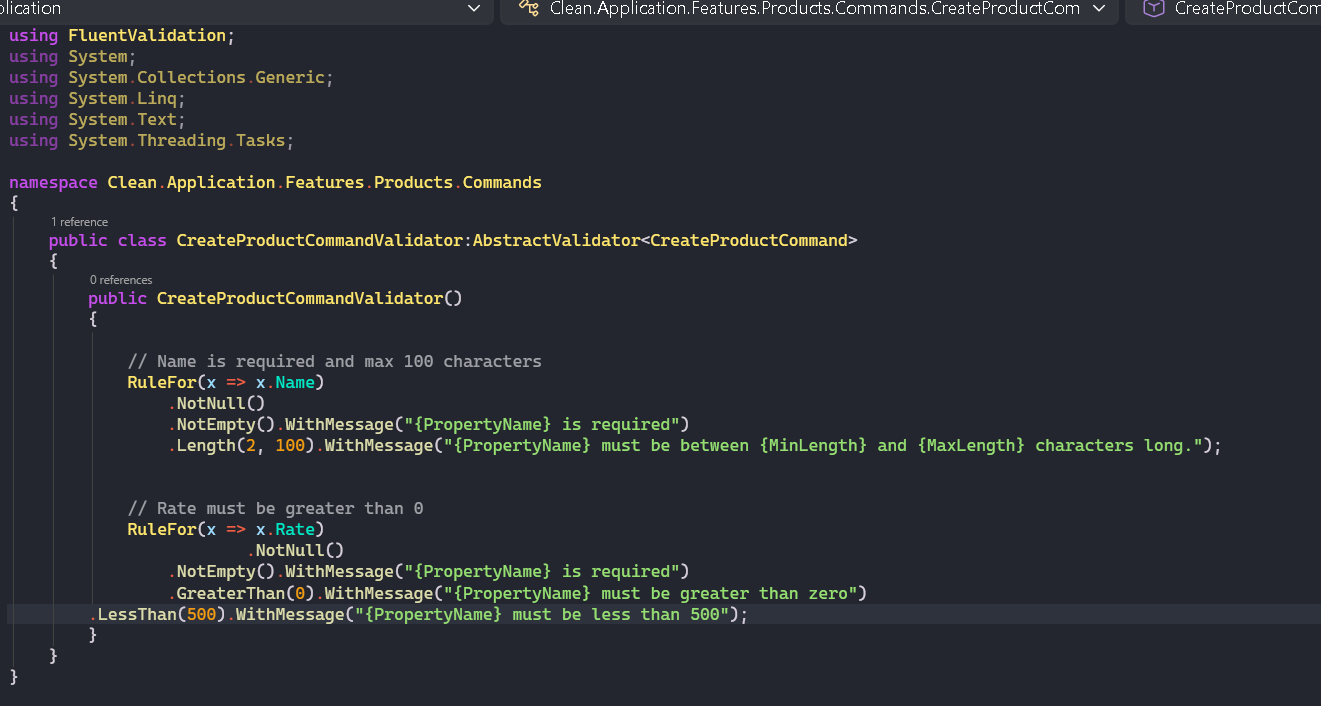
Install packages in application layer



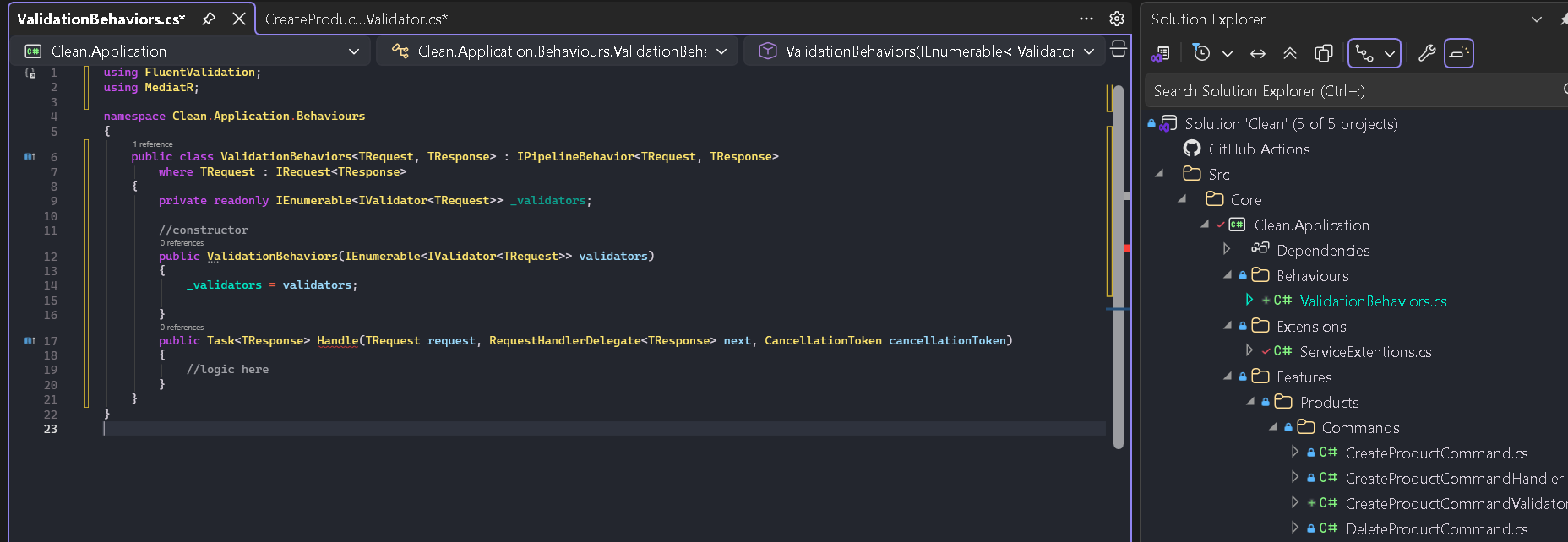
Register in service extension of application layer



Create validator for create command for now create as your needs



Now create behaviour



This is boilerplate logic added accordingly eg below