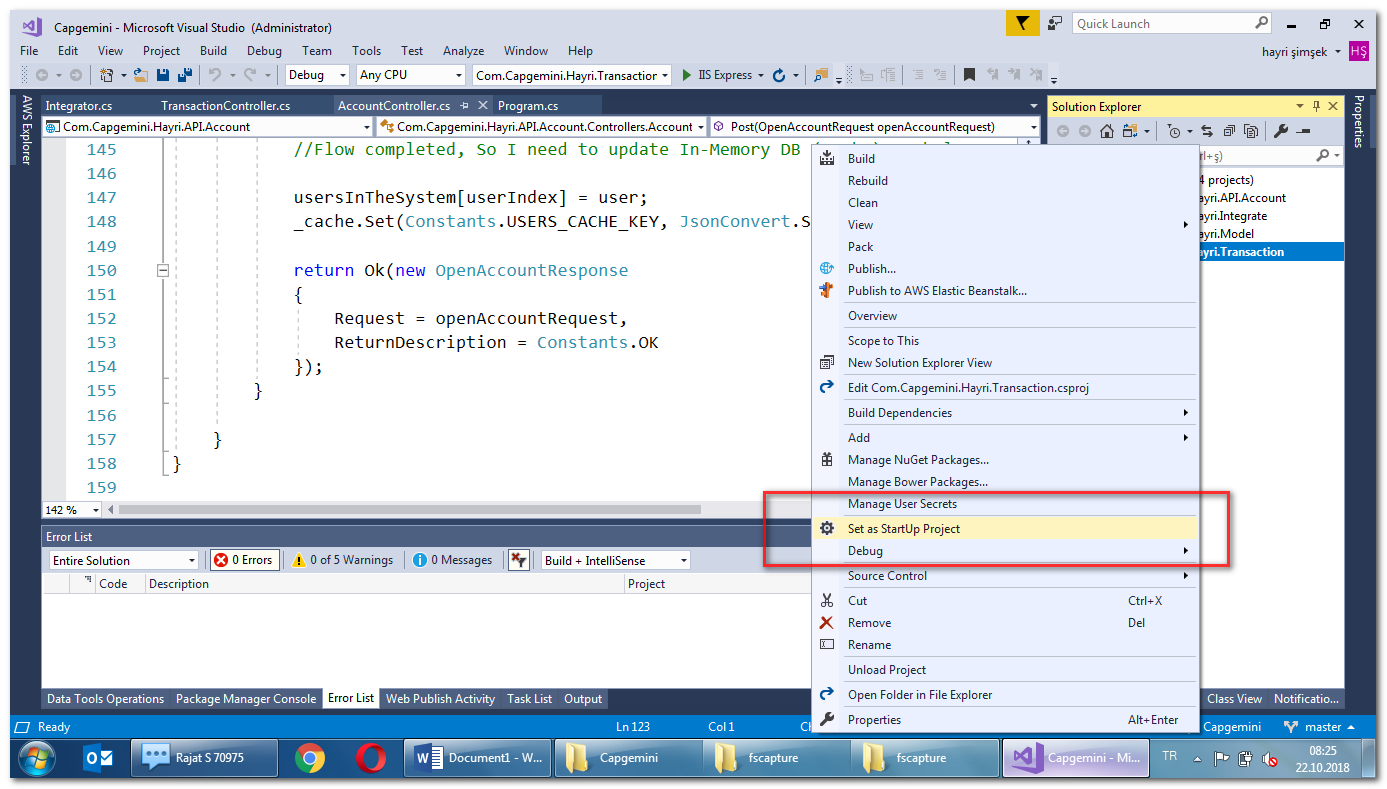
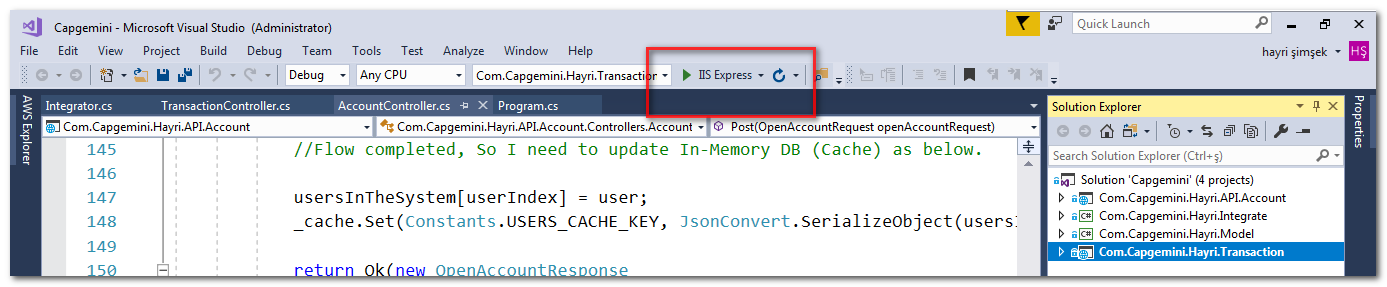
**Blue Harvest Code Assignment**

In order to test this solution; first, transaction API must be run as can be seen in the below images.

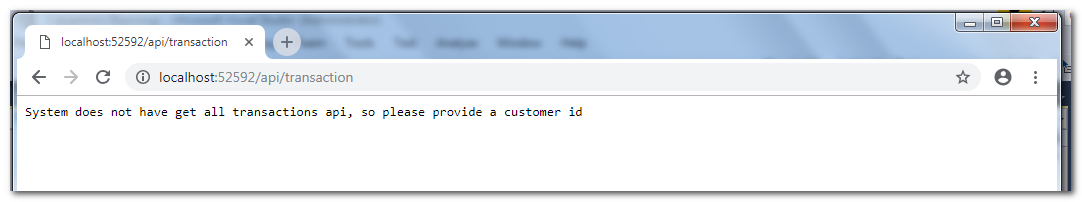
1.



2.

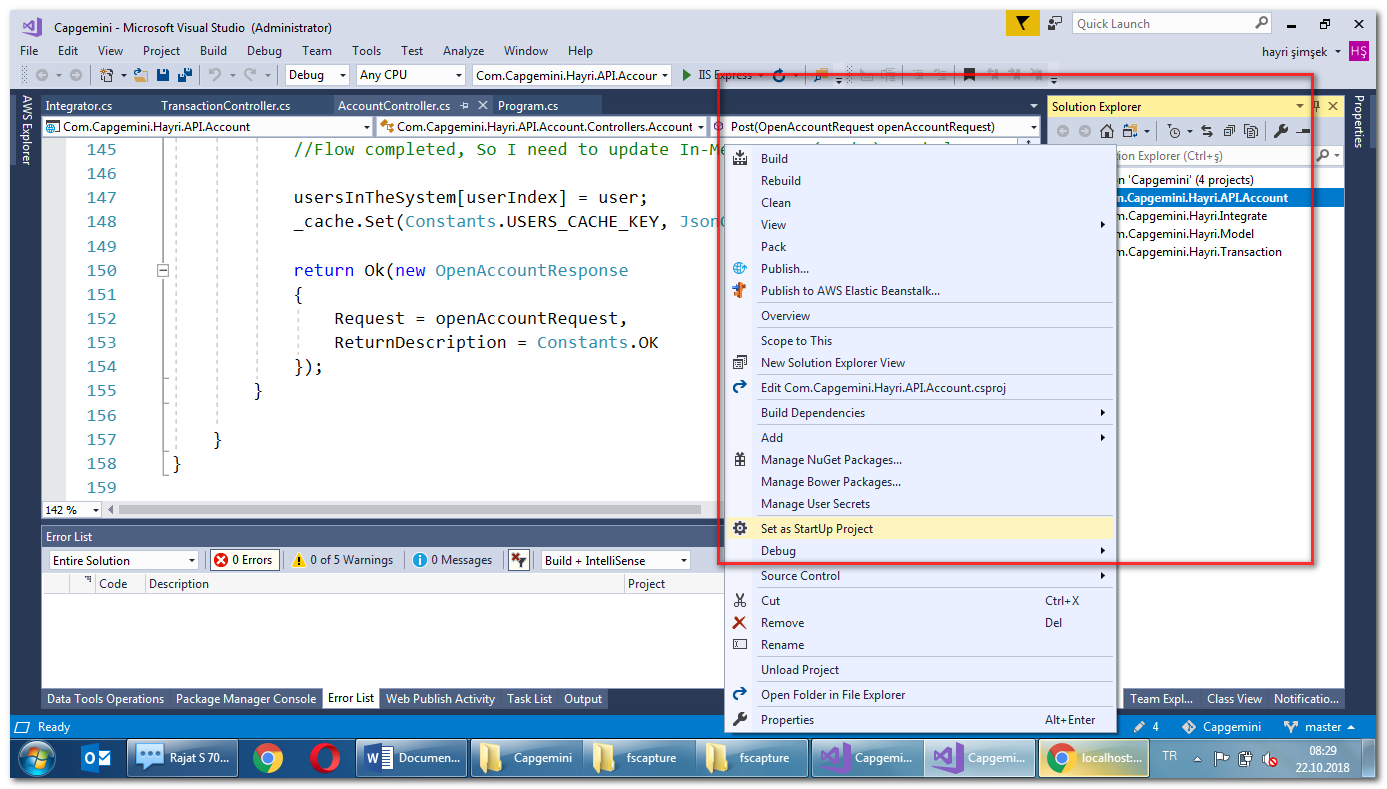


3. you should see something like below

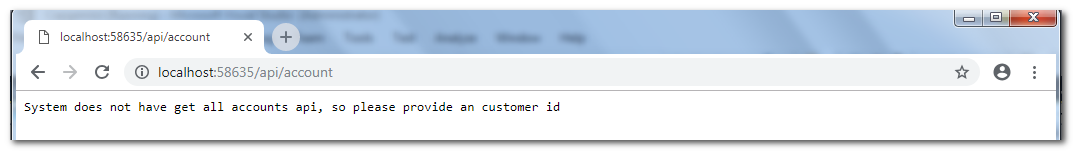


4. Now you need to open another Visual Studio and perform the same operations for Account API.

PS. First Visual Studio should not be closed

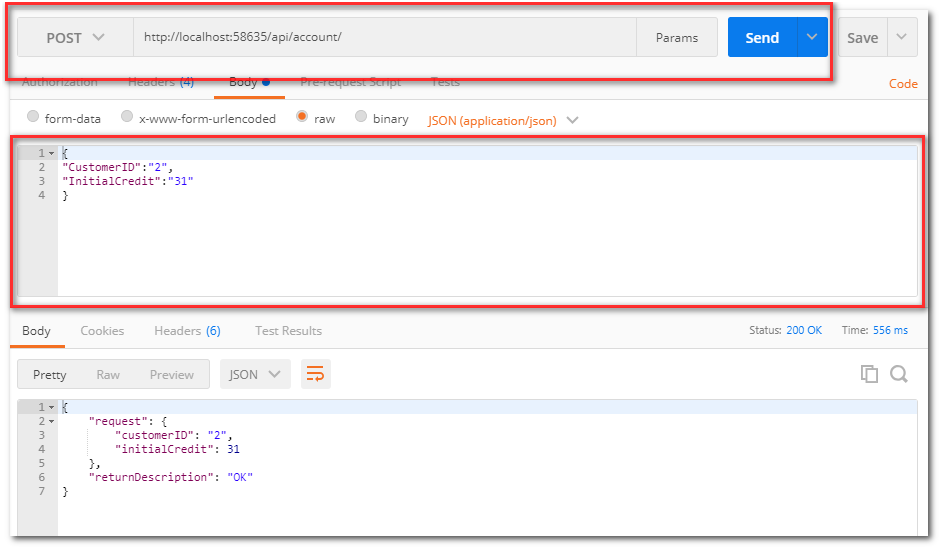


5. You should again click play button as in step 2. Finally you will see someting like below.

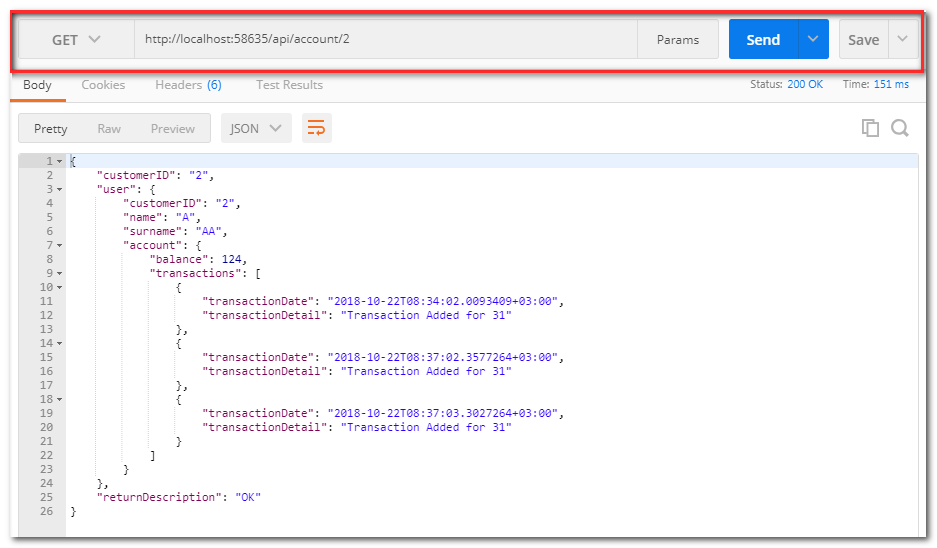


6. Now we have two APIs ready. In order to test them we can use a REST tool like POSTMAN.

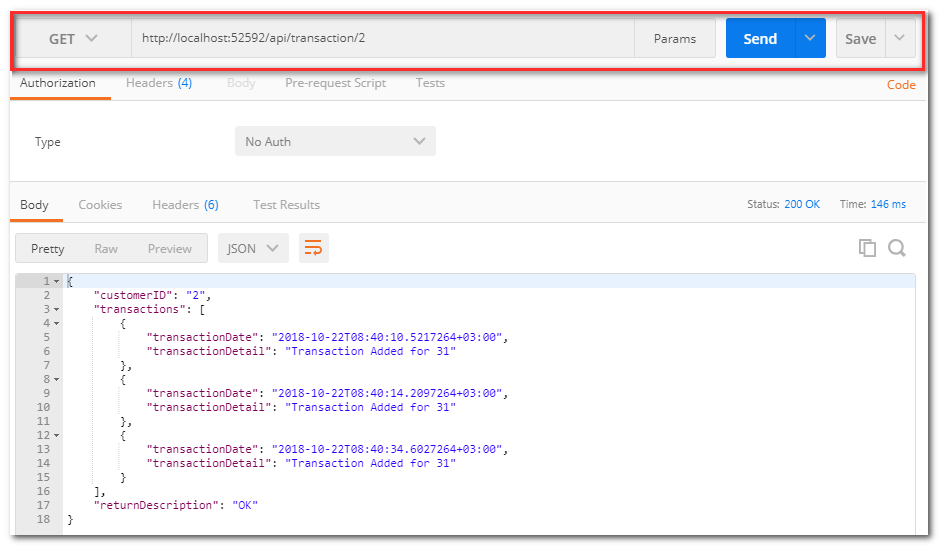
- POST Account: API end point and request should be like in the below image. You can call this API several times with same parameters or you can modify parameter values in the request. Of course, request model must be same as in the example.



* GET Account by id: In the below example, I tried with CustomerID=2



7. You can also directly call transaction API with the parameters below.



PS. As mentioned in the task, I have two APIs with different ports.

* Account API: Responsible for creating account and triggering transaction API. In this code, I am triggering transaction API via HTTP. In real case scenario, this would be via a message quee. In this API I am using microsoft standard in memory cache libraries. In real worl scenario, I would use REDIS since it supports SET and Lists
* Transaction API: This is responsible for creating a transaction for given Customer ID. For simplicity, I am not calling account API to check if this user already exists or not. In this API, I am using microsoft standard in memory cache libraries. In real worl scenario, I would use REDIS since it supports SET and Lists. Since first API and second API is different services, there is no connection between transaction API cache and account API cache. So, If you close one API, there will be some inconsistencies amoung caches.