ShopBridge: WebAPI Details

Vignesh Ponnuvel, vigneshponnuvel@outlook.com 10-07-2021

ShopBridge: WebAPI and Database Design for the activities:

1) Add Item to the Inventory
2) Modify an item in the Inventory
3) Delete an item in the Inventory
4) View the items in the Inventory

The solution has been prepared using .Net Core Web API. Database used: MS SQL Server. Please find the steps to run the solution along with sample screenshots.

Open the document DB_Structure and run all the scripts in MS SQL Server instance. Once done, please open the solution from the github [https://github.com/vigneshponnuvel/tB_API] - shopBridge.sln.

Update the DB connection strings accordingly in the files:

1. ItemDataProvider.cs in DataProvider folder
2. SHOPBRIDGECOntext.cs in Models folder
3. appsettings.json file

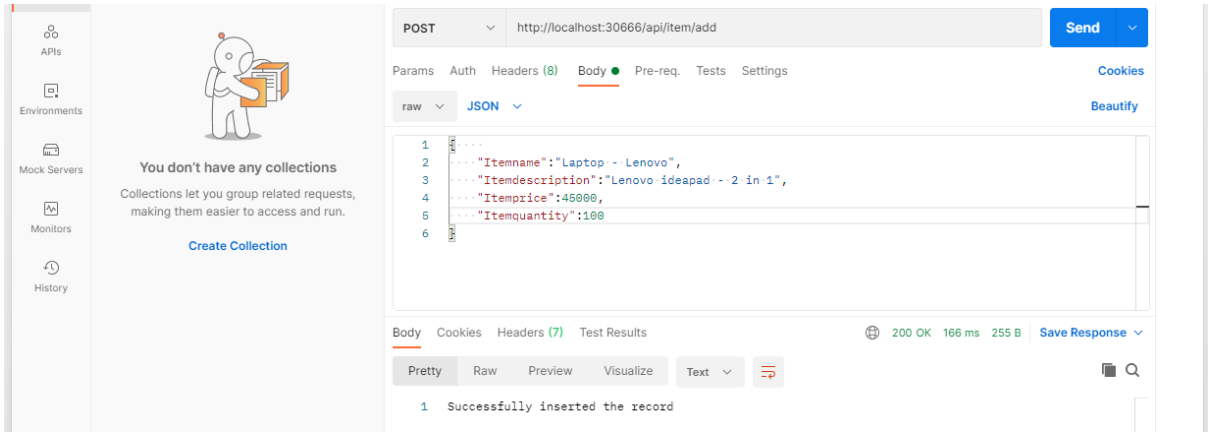
Build the solution and then run it. The browser will open with the address *localhost:30666/api/item/view* [the port may vary]. Since there will be no items in the database, it'll display [].

Now, use Postman or any other API testing tool and add/update/delete items in the Inventory table. Please find the below screenshots for reference:

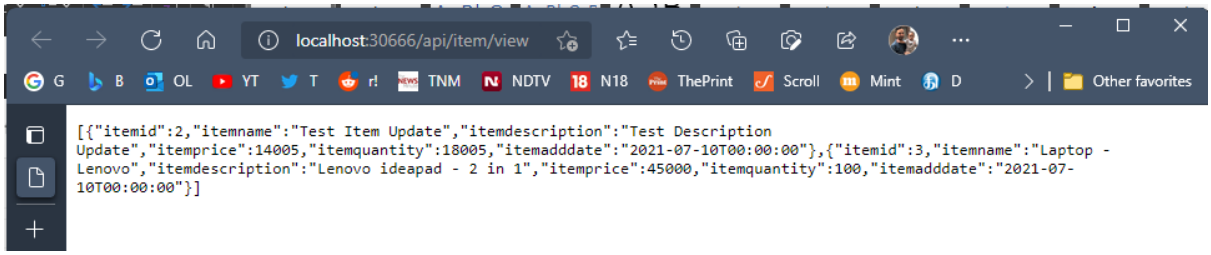
1) Add Item (URL: *http://localhost:30666/api/item/add*)

   JSON Input format:

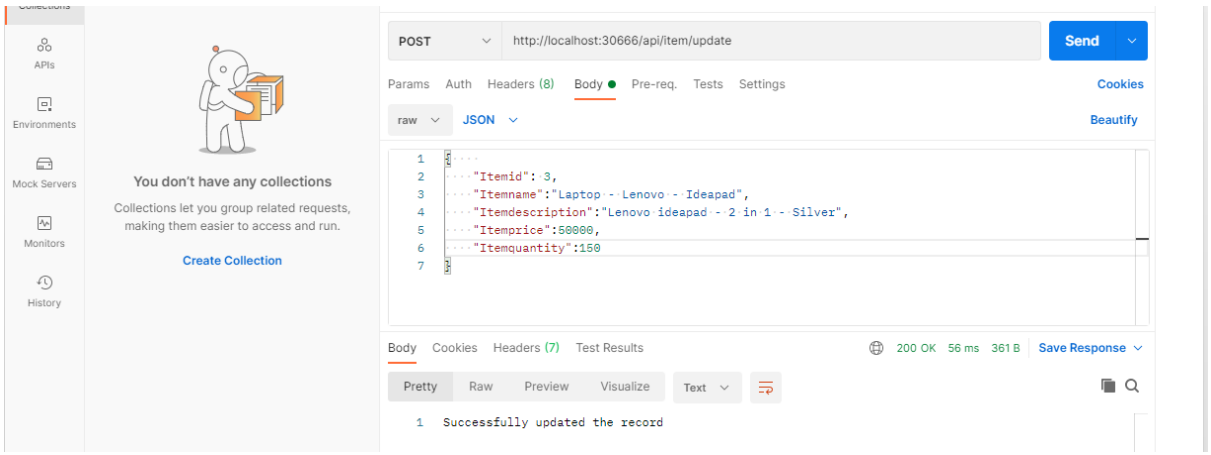
```
{
    "Itemid": 3,
    "Itemname":"Laptop - Lenovo - Ideapad",
    "Itemdescription":"Lenovo ideapad - 2 in 1 - Silver",
    "Itemprice":50000,
    "Itemquantity":150
}
```



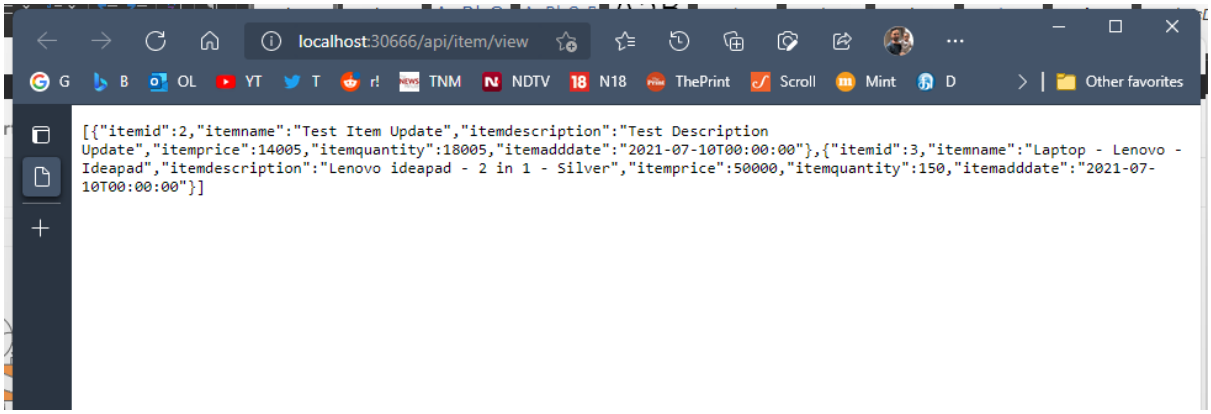
2) On successful addition, refresh the browser and the added details will be displayed.



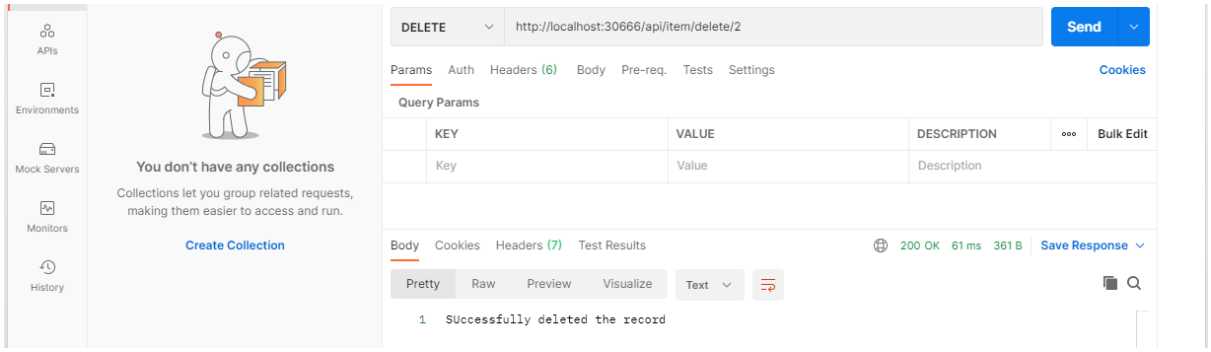
3) Update an item: (URL: *http://localhost:30666/api/item/update*)

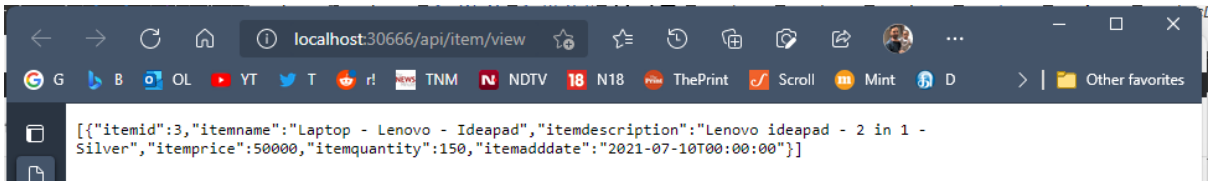


4) Refresh the browser to view the changes



5) Delete an item: (URL: *http://localhost:30666/api/item/delete/2*)

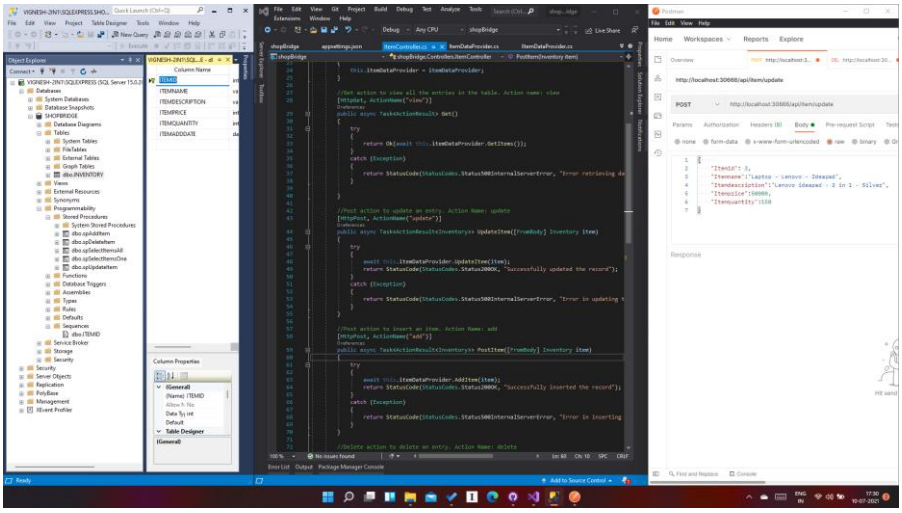




Additional Details:

IDEs Used:

1) Visual Studio 2019
2) MS SQL Server (SQL Express Management Studio 18)
3) Postman



Technology/Nuget Packages used:

1) .NetCore 3.1
2) Dapper
3) Entity Framework Core
4) SQL Server

Timelines:

1) Database Design and Execution: 45 Minutes
2) WebAPI (including EF) : 2 hours 30 minutes