PROG36944 - Advanced .NET Server Side Technologies
Assignment 2, Winter 2021
Instructor: Mahboob Ali

Scenario
Congratulations! You have been given a lucrative contract to build the ASP.NET Core MVC Web Site to manage products by their categories.

The basic idea of this application is to design a simple Product Registration System, where you should have a proper **Welcome page**.

Instructions

Hint: Follow the Example from the class like API Example.

1. Create a Web API Project named: [YOUR_INITIALS]_LAB2 (ex. MA_LAB2) that:
    a. Act as a **DBA** and Create tables in **database** named **Product**
        i. **Product:** ProductId, Name, Price, CategoryId
        ii. **Category:** CategoryId, Name
    b. Act as a DBA and insert two categories for the **Category** for testing purpose.
    c. Act as a DBA and insert 5 products for **Product** for testing purpose.
2. Add your model classes to your project using **Database First** (reverse engineering) from **Product** database. Use the link below to get some guidance and the command you need for the reverse engineering
https://docs.microsoft.com/en-us/ef/core/managing-schemas/scaffolding?tabs=dotnet-core-cli
    a. Analyze your project after the command, model folder, migration folder
3. Use Data Annotation Attributes appropriately to assist with the validation and display of your model. (apply data annotation on the model classes)
    a. Update the page titles, application name and page footer to reference the name: "[YOUR_INITIALS]'s Web API Product Registration System" throughout the site. For example, I would use: "MA's Web API Course Registration System"
    b. Updates the Navigation Bar to include links to the Product and Category Index views
4. Create Web API Controllers for the Product and Category entities in the model.
5. Create the CRUD operations for the API, to provide functionalities for
    a. Adding a product and category
    b. Reading a product and category
    c. Updating a product and category
    d. Deleting a product and category
    e. PATCH a product and category
6. Use **Postman** to test the CRUD operation of your API.
7. Add support for WEB API help documentation for all your controller actions, using SwashBuckle.
8. Capture the following screenshots and include them with your submission (see below):
    a. Database script for both the tables.
    b. Tables after the testing data has been entered.
    c. Scaffold command for the reverse engineering.
        i. Project solution explorer snapshot with new model and migration folders and files in it.
        ii. Each model classes.

d. Data annotations on the model classes.
e. API Controller
f. All the functionalities in the API
g. Postman screen shots for all the tests
    i. GET
    ii. POST
    iii. PUT
    iv. DELETE
    v. PATCH
h. API documentation in the browser

Additional Requirements

- **Submit your assignment (including all screenshots) via SLATE as per the submission requirements below**

Submission Requirements

- Include a few appropriate screenshots and put them in a Word document named as follows: [LASTNAME]_TLAB1.docx.
- **Zip up your** complete project and name it: [LASTNAME]_TLAB1.zip.
- **Make sure** you have captured all your screen shots.

GOOD LUCK!