Continue with the ASP.NET Core MVC Catering Management application from Part 3.
Please use the posted solution for Part 3 as the starting point for Part 4.

We are about to hire staff to help maintain the data in the database. In this part we will be adding Authentication and Authorization using Identity to control who can access the application and what they are allowed to see and do once they are in.

Instructions:

1. Configure the Identity system and add a class to seed data for Roles and Users. You will call it to run from Program.cs.
   - a. Make sure that both DbContexts are using the same database.
   - b. If you are using .EnsureDeleted() in your Initializer code, make certain that you only delete the database once.
2. Use the Seed to do the following:
   - a. Create roles called “Admin”, “Security”, “Supervisor” and “Staff”
     - i. Create user admin@outlook.com with password "Pa55w@rd" and add it to both the Admin and Security roles.
     - ii. Create user security@outlook.com with password "Pa55w@rd" and add it to just the Security role.
     - iii. Create user supervisor@outlook.com with password "Pa55w@rd" and add it to the Supervisor role.
     - iv. Create user staff@outlook.com with password "Pa55w@rd" and add it to the Staff role.
   - b. Create user user@outlook.com with password "Pa55w@rd”, not in any role.
   - c. Also create three additional users with password "Pa55w@rd”, one for email jkaluba@niagaracollege.ca, one for dstovell@niagaracollege.ca and one for your own College email address.
     - i. Add each of these users to both the Admin and Security roles.
   - d. **Note:** Make sure you set `EmailConfirmed = true` for each IdentityUser as you create them!
3. Modify access to the application as follows:
   - a. Anyone can access the home controller (all actions)
   - b. A user who is logged in but not in any role can also:
     - i. View the list and details of both Functions and Customers but cannot download documents.
   - c. A "Staff" user can also:
     - i. Create and Edit Functions, including uploading documents.
     - ii. Create new Customers.
       1. Edit a Customer only if they created the record.

    iii. View the full list of uploaded Function Documents as well as Edit and Download documents. However, they cannot Delete them.

d. A " Supervisor " user can also:
    i. View, Create, Download and Edit all data.
    ii. Can fully maintain all Lookup values.
    iii. Can view and download all reports.
    iv. Can delete an uploaded Function Document.
    v. Can delete a Function only if they created it.
    vi. Cannot Delete Customers

e. An "Admin" user can do anything except assign users to roles.

f. A “Security” user is essentially like a user who is logged in but not in any role except that they will be able to assign users to roles in the system.

Do NOT hide any links based on Authorization restrictions at this point.

Clarification for Authorized Access…

	INDEX	DETAILS	CREATE	EDIT	DELETE
Functions					
User (Can see file names but can’t Download Documents)	✓	✓			
Staff	✓	✓	✓	✓	
Supervisor	✓	✓	✓	✓	✓*
Admin	✓	✓	✓	✓	✓
Customers					
User	✓	✓			
Staff	✓	✓	✓	✓*	
Supervisor	✓	✓	✓	✓	
Admin	✓	✓	✓	✓	✓
Lookup Values (All), Reports					
User					
Staff					
Supervisor	✓	✓	✓	✓	✓
Admin	✓	✓	✓	✓	✓
Function Documents					
User		NA	NA		
Staff	✓	NA	NA	✓	
Supervisor	✓	NA	NA	✓	✓
Admin	✓	NA	NA	✓	✓

*Only if they entered the record into the system.

4. Add a UserRole controller and views that will only allow users in the Security role to add and remove users from roles. You can either use checkboxes or two multiselect list boxes for adding or removing roles from a user. Make sure there is a link in the navigation menu for this new controller.
   a. BONUS: No user should be allowed to change their own roles!

You will **not** hand in Part 4A. This work is in preparation for the next steps in Part 4 on the MVC Catering Management application.