Steps for creating an ASP.NET MVC Core project

1. Create New Project
   1. Under Visual C#, select Web
   2. Click on ASP.NET Core Web Application
   3. Give project a name and location to be saved
2. On the New ASP.NET Project window, select the “Web Application (Model-View-Controller)” template. Confirm you are using “.NET Core” in the top left and select “ASP.NET Core 3.+” as the version. Also, click on “Change Authentication” and select “Individual User Accounts”.
3. In Solution Explorer, right-click on the Models folder, and then add a new class. This will be your first model.
4. In the class file that you just added, give it any properties that the object may need. Remember, this model will represent a table in your database and the properties represent the columns of that table in the database.
5. In the Data folder, click on ApplicationDbContext.cs. Under class ApplicationDbContext, add a property of type DbSet<> i.e. DbSet<Person> People
6. For migrations, locate the Package Manager Console. Then type in the following commands:
   1. Add-Migration EnterYourMigrationNameHere
   2. Update-Database
7. **To add a Controller without (w/o DbContext) scaffolding:** 
   1. right-click on the Controllers folder and click Add 🡪 Controller. Then select “MVC Controller with read/write actions”
   2. Give the Controller a name
   3. Create a new folder in the Views folder and name it the same name as your controller. i.e. Controller Name: PeopleController, New Folder Name: People
   4. Right-click on the subfolder and click Add 🡪 View.
   5. Give the View a name i.e. Create. Then, select a View template from the drop-down list i.e. Create
   6. Select the drop-down list for Model class. Select the Model you are creating a Controller for. Also, select the data context class you want to use. It will likely be ApplicationDbContext.
8. **To add a Controller with scaffolding:** 
   1. Right-click on the Controllers folder and click Add 🡪 Controller. Then select “MVC Controller with views, using Entity Framework”. This is called scaffolding.
   2. When the Add Controller window pops up, select the drop-down list for Model class. Select the Model you are creating a Controller for. Also, select the data context class you want to use. It will likely be ApplicationDbContext. Leave all three boxes checked under the Views section. Then click “Add”.
9. To check out your Connection String, double-click on the appsettings.json file in the Solution Explorer
10. **To use the ApplicationDbContext inside a controller**
    1. If the ApplicationDbContext is not already inside of your controller, create a global private readonly variable of type ApplicationDbContext at the top of the controller class.
    2. Add a parameter of type ApplicationDbContext to the controller’s constructor.
    3. Inside of the constructor set you global ApplicationDbContext equal to the parameter ApplicationDbContext.
    4. The ApplicationDbContext is automatically set up for Dependency Injection. At application runtime it will inject and instantiation of the ApplicationDbContext into the constructor and initialize your global AplicationDbContext variable. You will then be able to use it inside your action methods and have access to your database.