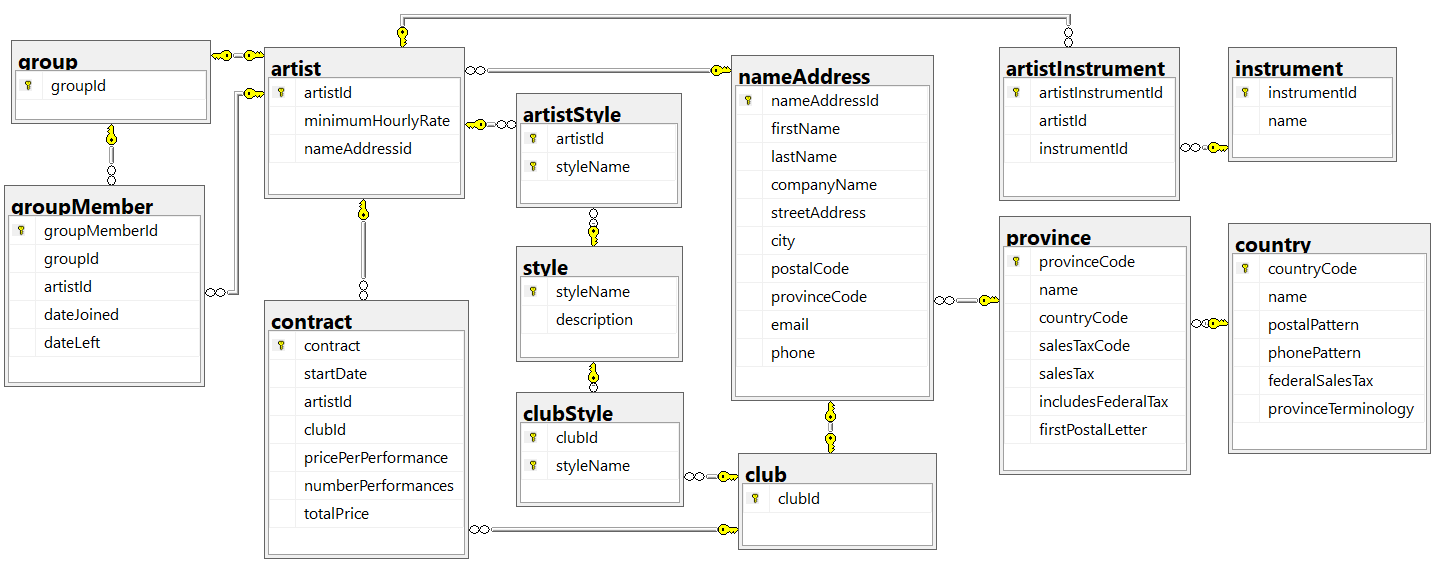
## Ms Web Tech Assignment 1 – MVC Basics

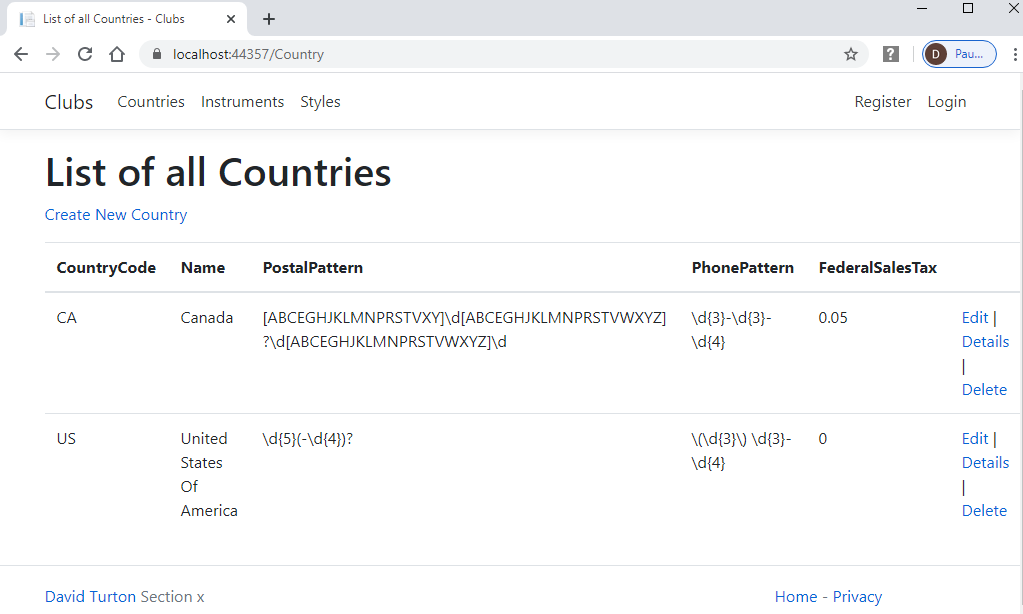
In this assignment, you are working on a site to record contract information between a clubs and their artists. You need to install the database from a script, generate the models and create views to display and maintain the leaf (independent) tables ***country, instrument,*** and***style***. In this assignment and all subsequent assignments, whenever you see “XX”, replace it with your initials.



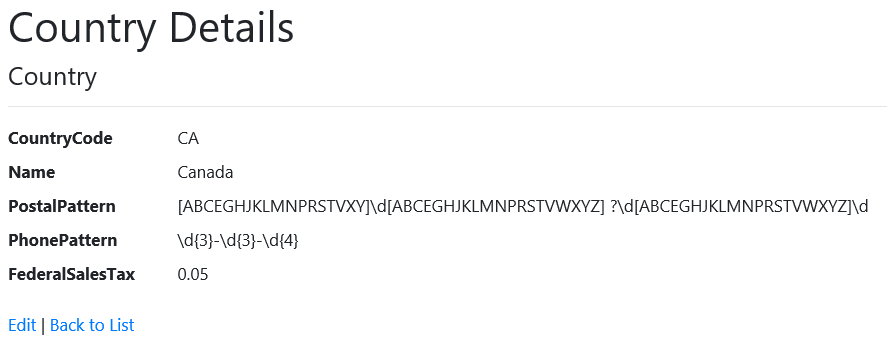
## XXClubs project

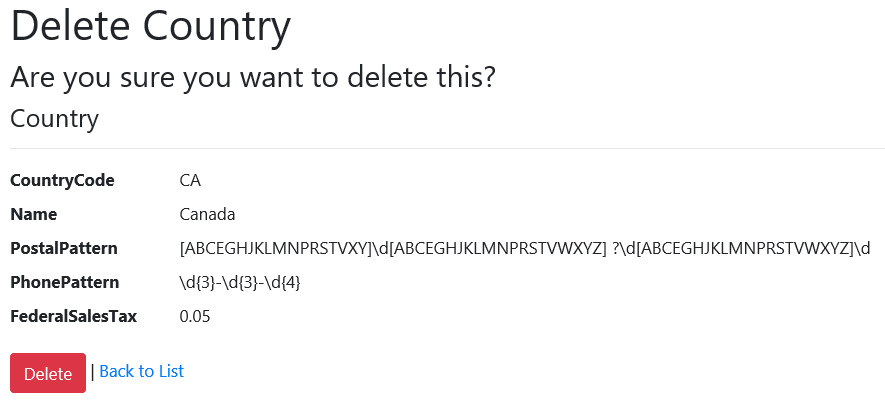
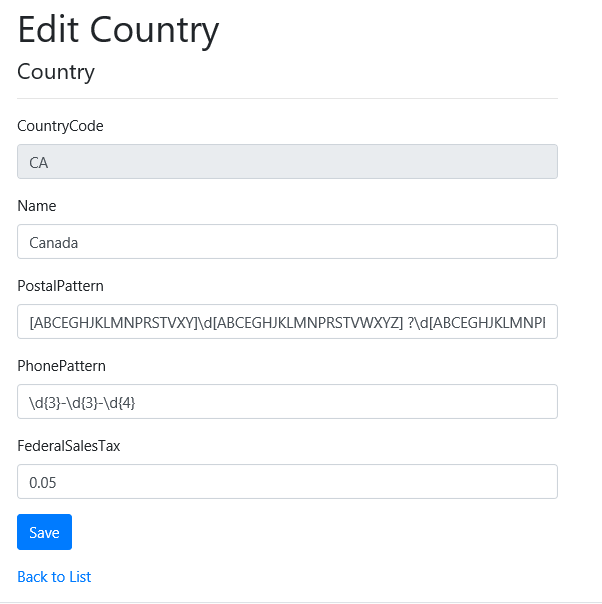
1. ☑ Create an ASP.NET Core 3.1 project using MVC. Call it ***XXClubs*** (where XX are your initials).
   1. Remember to change the authentication to ***Individual User Accounts*** … otherwise, Entity Framework support isn’t preloaded for database access, and is a pain to add later.
   2. Enable Razor runtime compilation  
      <https://docs.microsoft.com/en-us/aspnet/core/mvc/views/view-compilation?view=aspnetcore-5.0&tabs=visual-studio>
2. Load the database to SQL, then generate the models & context for Entity Framework:
   1. Download the ***Clubs.sql*** database script from the course site
   2. Use SQL Server Management Studio (SSMS) to execute the script and install the database.
   3. Generate the Entity Framework classes for the database into the project’s *Models* folder. Name the context ***ClubsContext*** (which is the default for a database named “Clubs”).
   4. Add the connection string to ***appsettings.json*** and add a service to ***Startup.cs*** to support dependency injection for database context ***ClubsContext*** (see the course notes for the first week).
3. Modify the default layout’s footer:
   1. Put your name and section on the left side of the footer. Your name should be a hyperlink causing the user’s e-mail client to create an email directed to your college e-mail account. Email subject should be Email from **XXClubs.**
   2. Move the menu’s hyperlinks for ***Home*** and ***Privacy*** from the menu bar to the right side of the footer. They should both be on the same line as your name.

## XXCountryController



1. Generate a controller called **XXCountryController** along with the required Views to enable full CRUD maintenance on the *country* table. Add a link to it on the menu.
   1. Normally, pages do not display keys. The key to this table is manually entered (and does not end in “ID”), so Razor provides it as an input on the Create view.
   2. Show the *CountryCode* on all views for this controller, but only allow the user to modify it on the Create view. Ensure all views & hyperlinks function correctly: Edit is particularly sensitive to incorrect changes.
2. Change the browser window title and the heading on the Index view to “Countries on File” or “List of Countries” instead of “Index” … less ambiguity for the user this way.
   1. Change the browser titles and headings on the Details, Create, Edit and Delete views as well, to denote the table and action (“Edit Country”, and so on).





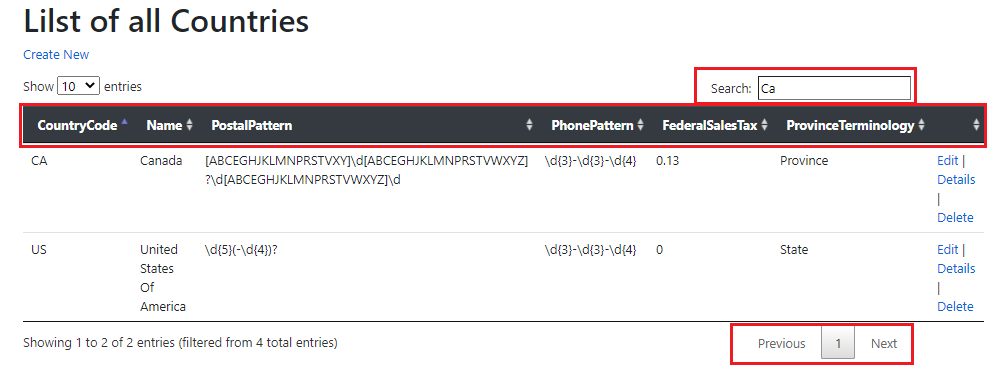
## XXStyleController, XXInstrumentController

1. Generate controllers for the ***style*** table, and the ***instrument*** table … with the required Views to enable full CRUD maintenance.
   1. Add a link for each controller to the menu.
   2. Where the key is manually entered (styleName), display the key on all views, but only permit it to be modified on the Create view. ***Do not display the autonumber key in instruments.***
   3. Modify the browser title and page headings on each view to indicate the object type and the action performed by the page (“Create New Style”, “List Instruments on File”, etc.).
   4. Challenge: display the style’s description on edit as a 3x40 textarea. No credit … just fun.

## All Three Controllers

1. Ensure the controllers, their menu entries, and their actions all work.
2. Comments are required to describe what each controller does and what each of its methods do. Generated controllers have 9 methods (actions). At 1% each, that’s a 10% penalty for each controller. Don’t bother commenting Views and Models … they usually break when you do.

## HTML Tables

1. All html tables should have dark header.
2. Search, paging and sorting should be enabled.   
   

<https://datatables.net/>

## Hand In

1. Zip and upload your project folder to the eConestoga upload folder (course tools/assignments/Assignment 1). Only the last upload will be retained.