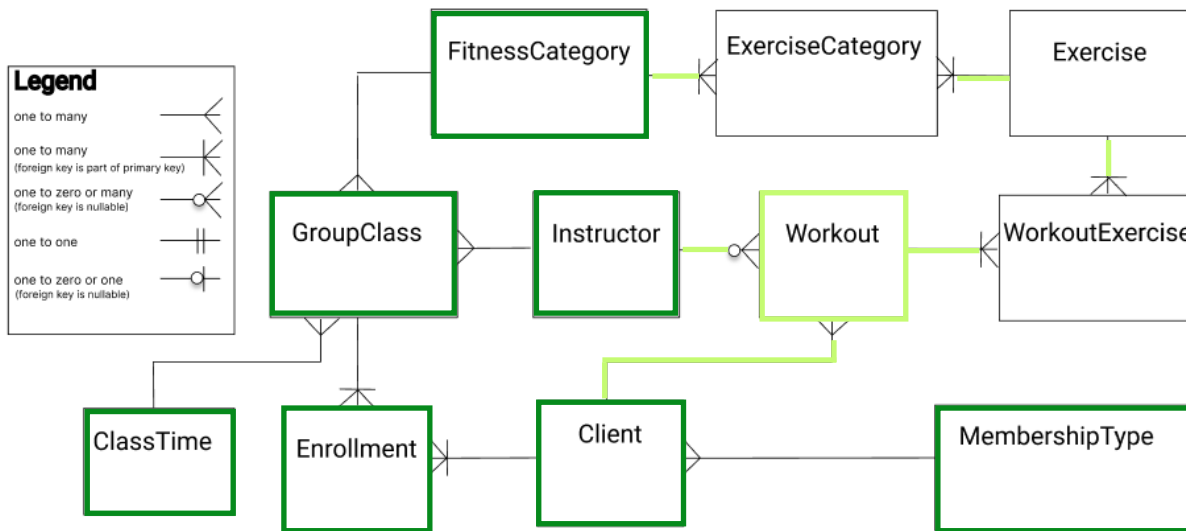## Gym Management Data Model

Instructions

Continue with the ASP.NET Core MVC Gym Management application from Part 2. If you had issues completing Project Part 2 then use the posted solution for Part 2 as the starting point for Part 3. Note that Model Classes for the additional entities shown above have already been added in the posted solution for Part 2.

Take time to examine the new classes and make sure you are familiar with the properties of each one.

1. (**2 Marks**) Make sure your name and student number are in comments at the top of your DbContext (GymContext) class as well as having your name displayed on the Home/Index Page.

2. (**8 Marks**) Seed data and DbSets for the new entities have already been added BUT you need to complete the following:
   a. The relationship between Instructor and Workout has not been created yet. Add properties to the Instructor and Workout entities to support the 1:0 or Many relationship.
   b. Only allow cascade delete for the following:
      i. If a Workout is deleted, all related WorkoutExercise records should be deleted as well.
      ii. If a FitnessCategory is deleted, all related ExerciseCategory records should be deleted as well.
      iii. All the other new relationships must restrict cascade delete.
   c. There are currently no migrations for the GymContext in the posted solution. Start a new set of migrations and make sure the initializer is set to use your migrations. Note that the supplied seed data should work with your model.

3. **(20 Marks)** We need to be able to upload documents related to the Instructor, including Certifications, Contracts, Resumes etc.

   a. **(10 Marks)** An Instructor might need many documents, but each document relates back to a single Instructor.

      i. If an Instructor is deleted, all related documents should be automatically deleted as well (Cascade Delete).

      ii. While creating or editing an Instructor, the user should be able to select multiple files at once and upload them to the Instructor.

      iii. The user should be able to click links to download the related documents.

      iv. The Instructor/Details view should also show a list of all uploaded documents that relate to the Function with links to download them.

   b. **(10 Marks)** Add a new controller for managing the uploaded documents with the following:

      i. Index View showing the file names of the Documents along with the name of the Instructor it relates to.

      ii. Add filters for:

         1. Instructor with an All option at the top of the select element.

         2. Characters in the File Name.

      iii. Add paging but always sort by file name.

      iv. The Index should also have links for both downloading and deleting the documents.

      v. BONUS: Add a Description property to the Instructor Document and allow the user to edit both the File Name and Description from the new controller. Of course, descriptions cannot be added until after files have been uploaded.

You will **not** hand in Part 3A. This work is in preparation for the next steps in Part 3 on the MVC Gym Management application.