

C# Cumulative Part 1

By: Christine Bittle

This cumulative project involves building a Minimum Viable Product (MVP) on the Teachers table of the provided School Database using ASP.NET Core Web API and MVC. Part 1 is the READ functionality.

MVP: Create, Read, Update, Delete on the Teachers Table.

Type	Description	File
-	A class which connects to your MySQL database	/Models /SchoolDbContext.cs
Controller	A WebAPI Controller which allows you to access information about teachers	/Controllers /TeacherApiController.cs
Controller	An MVC Controller which allows you to route to dynamic pages	/Controllers /TeacherPageController.cs
Model	A Model which allows you to represent information about a teacher	/Models /Teacher.cs
View	A View which uses server rendering to display a list of teachers from the MySQL Database	/Views /Teacher /List.cshtml
View	A View which uses server rendering to display a teacher from the MySQL Database	/Views /Teacher /Show.cshtml

How to submit

1. Use Visual Studio / git to push your work to a remote repository (Parts 1, 2, and 3 can be on the same repository)
2. Verify the repository:
 - a. contains the work you wish to submit (i.e. the files are there)
 - b. is public (if it is set to private, change it to public!)
3. Include repository github link as part of your assignment submission (Do not share the link or your work with anyone else)
4. Include evidence of your testing as a PDF with screenshots of your cURL commands and web pages

(Quantitative: 8 Marks): Implement the required MVP using ASP.NET Core:

- (2 Marks) An API which returns information on all teachers
- (2 Marks) An API which returns all information on a teacher
- (2 Marks) A web page that displays information on all teachers
- (2 Marks) A web page that displays all information on a teacher

(Qualitative: 8 Marks): Document your work with (a) descriptive:

- (2 Marks) Summary blocks for your API methods
- (2 Marks) Teacher Model Properties (Teacher.cs)
- (2 Marks) Variable names
- (2 Marks) Project .readme

(Testing: 8 Marks): Include evidence of the following testing:

- (2 Marks) Your API that Lists Teachers
- (2 Marks) Your API that Finds a Teacher by the Teacher's ID
- (2 Marks) Your web page that displays all teachers
- (2 Marks) Your web page that displays a teacher

(Initiative: 8 Marks): Earn **up to 8** Initiative Marks by improving on MVP

- (2 Marks) Error Handling when trying to access a teacher that does not exist
- (2 Marks) On Teacher/List.cshtml, search for a Teacher by their Hire Date within a range
- (2 Marks) Read functionality for Students
- (2 Marks) Read functionality for Courses
- (2 Marks) On /Teacher/Show.cshtml, display all Courses taught by that teacher (Hint: try a [ViewModel](#))
- (2 Marks) Significant Layout and Styling Adjustments to /Teacher/List.cshtml and /Teacher/Show.cshtml

RUBRIC

	0 Marks	1 Mark	2 Marks
Quantitative	Functionality not implemented	Functionality partially implemented / implemented with room for improvements	Functionality implemented, no improvements required
Qualitative	Documentation not included	Documentation partially included / included with room for improvements	Documentation included, no improvements required
Testing	Testing not included	Testing partially included / included with room for improvements	Testing included, no improvements required
Initiative	No initiative elements attempted	Initiative elements partially implemented / implemented with room for improvements	Initiative included, no room for improvements