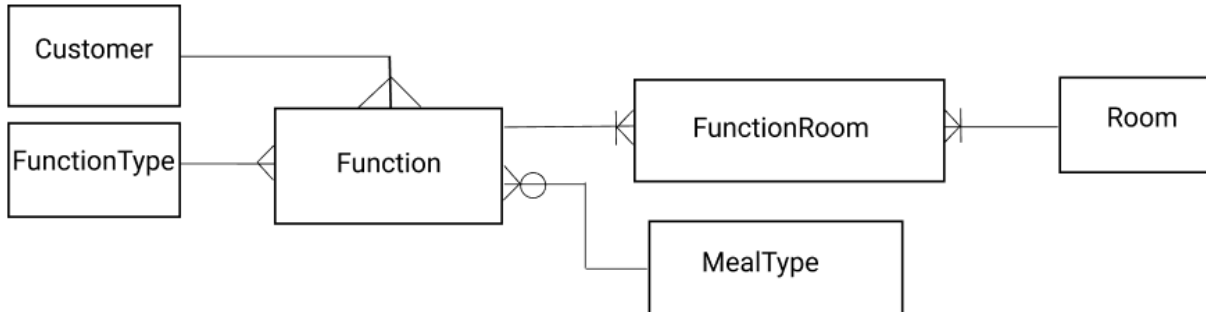


Instructions

Continue with the ASP.NET Core MVC Catering Management application from Parts 2A & B.



1. For both the **Function** and **Customer**:

- a. **(5 Marks)** Add Paging on the Index View with the option that the user can change the number of records displayed on a page.
 - i. User preference for Page Size must be remembered for each Index View separately.
- b. **(10 Marks)** Persist any Filter, Sort and Paging parameters for each Index View as the User navigates to Create, Update, Details and Delete and back to the Index.
- c. **(5 Marks)** Add Auditing to automatically maintain the four basic audit fields.
 - i. Use an Interface such as `IAuditable` to define the properties for Auditing.
 - ii. To audit who created or last edited a record, use the Identity of the Authenticated User if it is available.

2. For **Function** Only:

- a. **(10 Marks)** Add Concurrency Control for Edit Operations:
 - i. Use a Timestamp (`RowVersion`) to detect changes made to a record for Optimistic Concurrency Control.
 - ii. If there is a concurrency conflict, give as much feedback as possible to the user, showing the differences between the User's version of the record and what is in the database.
 - iii. Since we are using `sqlite` you will need to supply triggers to generate random BLOB values for the `RowVersion` on both Insert and Update.

Submit your work for Parts 2A, 2B and 2C together in one project.

As you submit to Blackboard, **you must enter comments describing how far you got** on the assigned work. Failure to describe the completeness of your submission will result in marks being deducted.