

Dish Cuisine Relationship - The dish cuisine relationship is one to one where one dish only belongs to one cuisine and one cuisine is only used to classify one dish. We are assuming that cuisine values have to be assigned to dishes specifically because it is connected to the dishId.

Dish Ingredients Relationship - The dish ingredient relationship is many to many where we assume many different dishes may contain the same ingredient and many different ingredients are part of the same dish.

Dish Rating Relationship - The dish rating relationship is exactly one to many where many ratings can belong to exactly one dish but each rating has exactly one dish it belongs to. We assume that each rating only is for one of the dishes.

Dish User Relationship - The dish user relationship is many to exactly one where each dish only belongs to exactly one user but each user can create multiple dishes.

User Rating Relationship - The user rating relationship is exactly one to many where one user can make many ratings but one rating is exactly made by one user.

User User Relationship - The user user relationship is many to many as we're assuming a user can follow multiple other users and those users can also follow multiple other users.

Entities:

- Dish. Each dish has a DishName, Calories and unique DishId.
- Ingredients. Each ingredient has IngredientName, Measurement, Price and unique IngredientId.
- User. Each user has UserFirstName, UserLastName and unique UserId.
- Rating. Each rating has Score and unique UserId and DishId.
- Cuisine. Each cuisine has CuisineName, Taste and unique DishId.

Tables:

- Dish(DishId: INT [PK], DishName: VARCHAR(255), Calories: INT)
- Ingredients(IngredientId: INT[PK], IngredientName: VARCHAR(255), Measurement: REAL, Price: REAL)
- User(UserId: INT [PK] ,UserFirstName: VARCHAR(255), UserLastName: VARCHAR(255))
- Rating(UserId: INT [PK] [FK to User.UserId, DishId: INT [PK][FK to Dish.DishId,Score: REAL)
- Cuisine(CuisineId: INT [PK], CuisineName: VARCHAR(255), Taste: VARCHAR(255))
- Madeof(CuisineId: INT [PK] [FK to Cuisine.Cuisine Id], DishId: INT [PK] [FK to Dish.DishId])
- Contains (DishId: INT [FK to Dish.DishId], IngredientId: INT [FK to Ingredient.IngredientId])
- Creates(UserId: INT [FK to User.UserId], DishId: INT [FK to Dish.DishId])
- Follows(UserId_1: INT [FK to User.UserId], UserId_2: INT [FK to User.UserId])

