Muscat College

Database Principles and Applications

Code: CSCUMB3

Assignment 2020

Submitted by

Stirling No:_2840080

Submitted on

27-Nov-2020

Database Principles and Applications

Part A

1) Entities and attributes:

Entity	Attributes	Justification
Member	Member ID	The boatyard firm keeps contact information including at least their name, age, and their e-mail address.
	Member Name	
	Age	
	Email	
Member Group	Member type	A member in a group may be a skipper or crew. So type means skipper or crew. There are at least 2 people in a group.
Booking	Booking ID	A group may do a booking for a yacht for a particular week in season, so the booking should have an ID.
	Week Number	A season is 20 weeks, each week is identified by a number.
Yacht	Yacht Name	Each yacht has a unique name. Length of yacht is recorded in meters. The price depends on the size of the boat.
	Length	
	Yacht Price	
Extra	Extra Name	Each extra is identified by a unique string and it is own price
	Extra Price	

2) Single table relation

Single table design of boatyard database in first normal form (1NF):

Boatyard Table
Member ID
Member Name
Age
Email
Member type
Booking ID
Week Number
Yacht Name
Length
Yacht Price
Extra Name
Extra Price

Potential Issues with this design:

To highlight the issues with using the database with a single table in first normal form, will load some data into the table and see how integrity is not respected:

Member ID	Member Name	Age	Email	Member type	Booking ID	Week Number	Yacht Name	Length	Yacht Price	Extra Name	Extra Price
1	Ali	26	ali@gmail.com	Skipper	1	Week 1	Isobel	12 m	900	blanket	50
2	Seif	24	sei@gmail.com	Crew	1	Week1	Isobel	12 m	900	blanket	50

In this data example, a booking to one group composed of 2 people: one is skipper and the other is crew, we can observe the redundancy of data:

- Yacht details are redundant.
- Extra name and extra price are redundant.
- Week number is repeated
- Booking ID is repeated.

The most important issue of using first normal form is the data redundancy, so the file size of database will be very important.

The second issue, is the maintenance of database, since it is difficult to do updates to the design even for the records in the table.

3) ER model

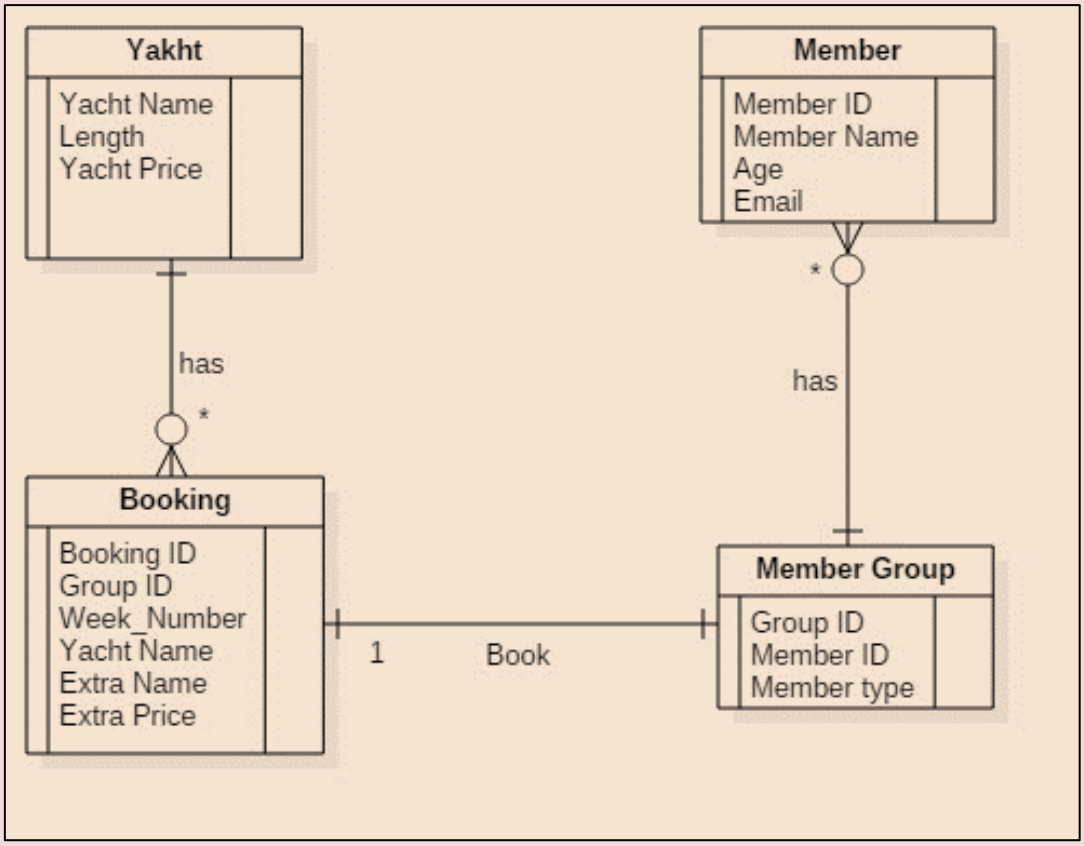

Figure 1

Resolving Many-to-Many

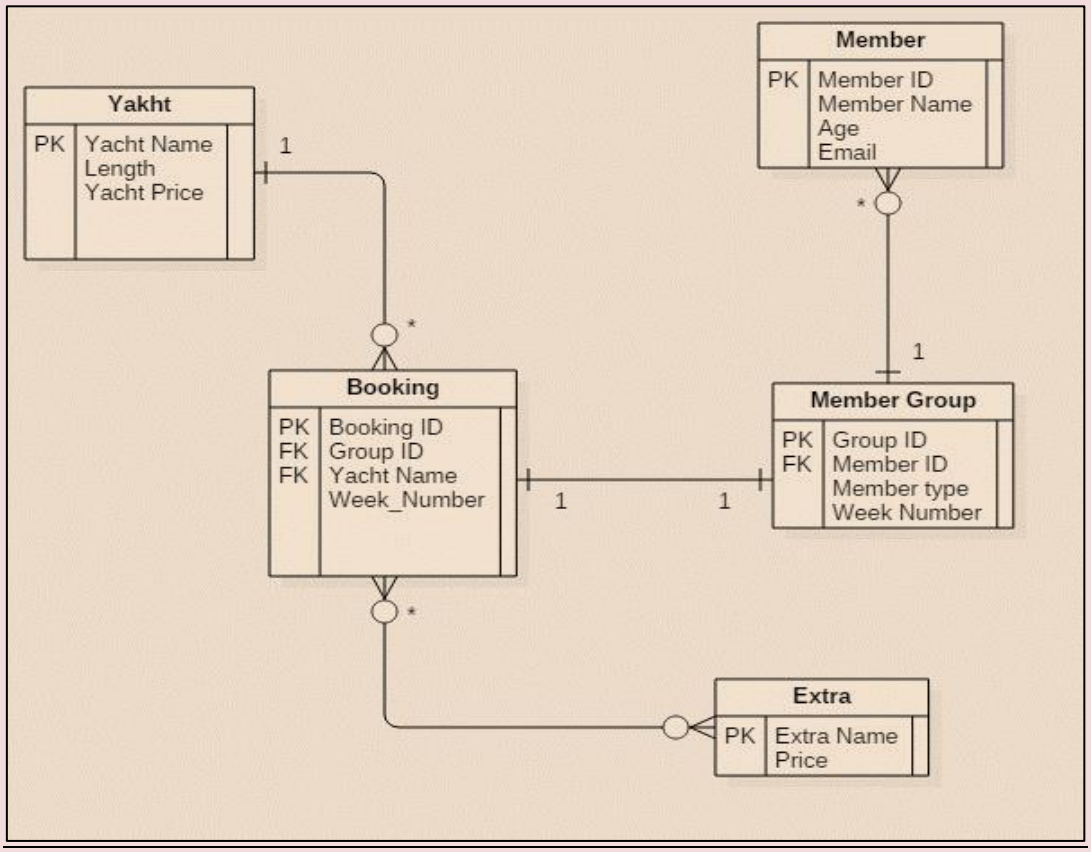

Figure 2

The ER diagram includes 5 entities: Yacht, Member, Member Group, Booking and Extra.

- Yacht: is an entity that encapsulate the yacht details.
- Member: is an entity to keep records of people who book the yacht.
- Member Group: is an entity that include the list of people in the group, and specify if the member is a skipper or crew. A Group should be referenced by a unique number, so the attribute Group ID is added.
- Booking: is an entity to record bookings of a particular group for a particular yacht. To identify a booking, “Booking ID” attribute is added.
- Extra: is an entity identified by a unique string and it is own price.

Relationships:

A Member can be assigned to many groups in different weeks, a Group contains at least 2 people so the relationship between Membre_Group and Member is [2 to Many].

A Group can book only one yacht a week, a yacht can be booked only for one group a week. So the relationship between yacht and booking is [one to one].

A yacht can be booked many times in different weeks, so relationship between yacht and booking is [1 to many].

Many-To-Many:

In database systems, Many-To-Many relationship cannot be accommodated. Therefore, before converting the database model in SQL server, we must switch from the Many-To-Many relationship to other relation.

As shown in figure number 2, the Many-To-Many relation among Booking and Extra is compatible to two One-to-Many relationships between Booking and Extra and a new entity that it is called Booking_Extras.

4) 3NF Model:

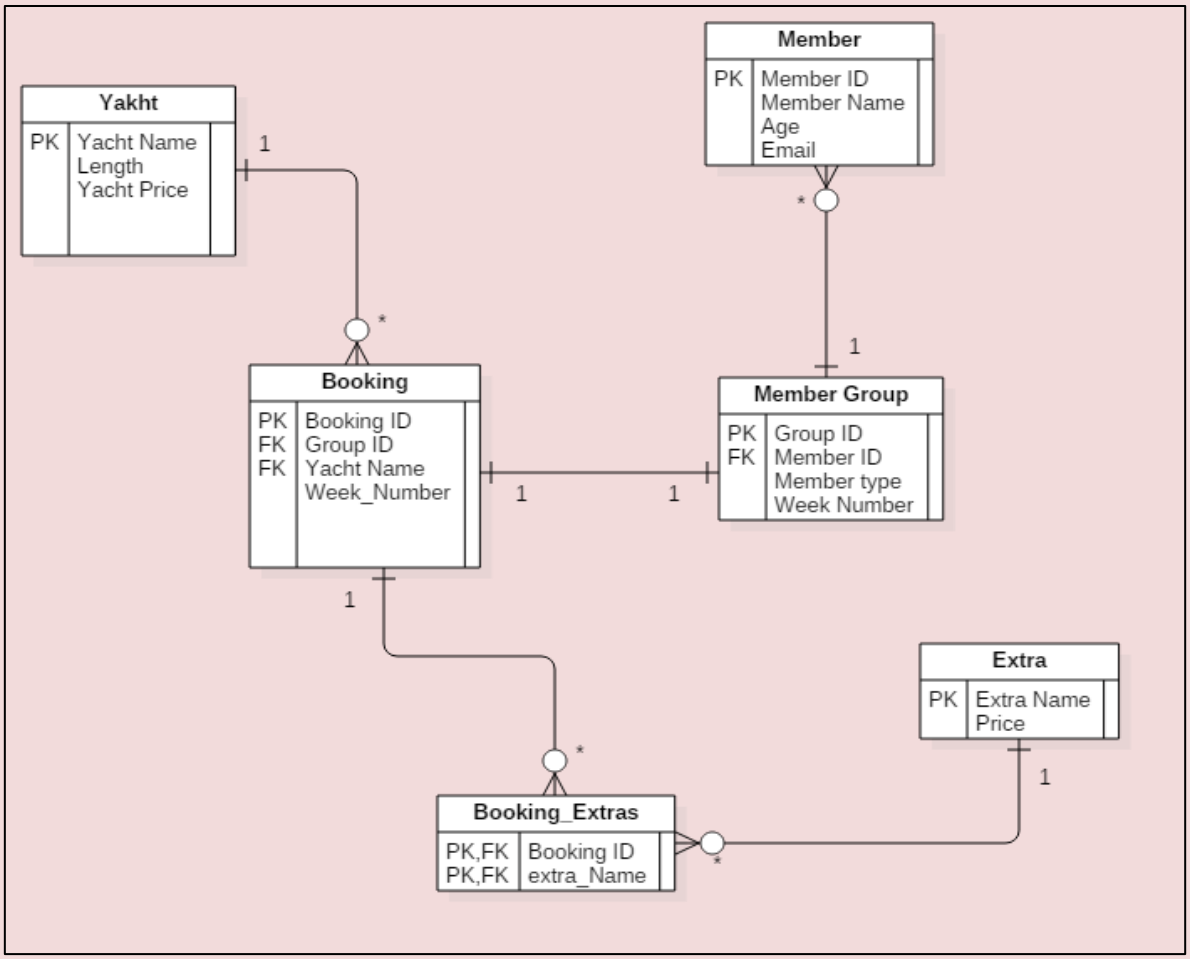

Figure 3

The new place which replaces the Many-to-Many relationship is called “intersection” it is intended that it is just an entity linking the Booking and Extra, if this relationship remains without solution or break, then there will be a complication or some problems in this company, for example, if you want to change something specific, you will have to go into ROWDATA table and change it from everywhere, so they must be separated.

So in figure number 3, I’ve turned Many-To-Many relationships by adding a new entity called Booking_Extras. Note that Extra_Name is the Primary Key in the Extra entity, and the

Booking_ID is the primary Key in the Booking entity, so in the new entity, I collected the two primary keys, as two foreign keys in the new entity.

- **Key words:**

Entities: are objects for anglings to tracked in the database.

Foreign Key: is the same as a PK (primary Key), but just located in a new place.

Primary Key: is the column that contains unique values in every row in the table.

Relationships: description of the entities how they will interact each other.

End.