This database model seems to be being used for tracking ferry cruises, passengers’ tickets for those cruises, the ports they travel to and from, as well as the cabin information and services available in those cabins. The breakdown of the tables is as follows:

Table 1 ‘Ferry’:

This table has an id field that is an incrementing integer which serves as the primary key for the table. This field is used by the Cabin and Cruise tables. It also contains string fields for the name and description of the ferries.

Table 2 ‘Cruise’:

This table also has an integer as primary key, which increments similarly to the Ferry primary key. It has a string field for the deck name, as well as integer fields for the cabin number, price, cabin type ID, and a farry ID which is a foreign key that indicated which ferry is used for the cruise.

Table 3 ‘Port’:

This table has a primary integer key (again, incrementing). It contains string fields for the name and address of the port. The port table is used by the ‘Cruise’ table to indicate the departure and arrival ports of the cruise.

Table 4 ‘Services’:

Each service has an incrementing integer primary key. The name and descriptions of services are stored in string fields of the table. Lastly, there is a foreign key for each service linking it to the relevant cabin type.

Table 5 ‘Cabin’

Much like the other tables, Cabin contains an incrementing integer value which serves as a primary key. It contains foreign key fields for the ‘cabin\_type\_id’ and ‘farry\_id’ fields, and integer fields for the Cabin number and price. There is also a string field for the deck (presumably the deck name).

Table 6 ‘Cabin\_type’:

This table has also has an incrementing primary key, string fields to store names and descriptions, and an integer place count field.

Table 7 ‘Tickey’:

The last table links the passenger to their relevant cruise and cabin, and contain a field for a binary QR code. Like the others, there is a ticket ID stored in a unique incrementing integer which is the table’s primary key.

SELECT

Cruise.id AS cruise\_id,

PortDeparture.name AS departure\_port,

PortDeparture.address AS departure\_address,

PortArrival.name AS arrival\_port,

PortArrival.address AS arrival\_address

FROM

Cruise

JOIN

Port AS PortDeparture ON Cruise.port\_departure = PortDeparture.id

JOIN

Port AS PortArrival ON Cruise.port\_arrival = PortArrival.id

JOIN

Services ON Cruise.id = Services.cruise\_id

JOIN

Ferry ON Cruise.ferry\_id = Ferry.id

WHERE

PortDeparture.name = ‘Charlevoix’

AND PortArrival.name = ‘South Haven’

AND Services.name = ‘Cocktail Service’

I think the ‘Port’ table is, for the most part, written correctly. However, there are likely more tables or fields that could be added to it to add more information. An example would be a field specifying the number of moorings.