In this schema, the diagram illustrates each entity and relationship required by Mrs. Patterson in the letter. I developed four entities – *Customer, Reservation, Room*, and *Type* which are respectively connected by the relationships *Makes,* *Contains,* and *Has.*

The *Customer* entity’s identifying attribute is “CID,” which is defined by their serial place in the reservation queue. Following that, the attribute CName contains the customer’s name for identification purposes. Also, as requested, I added “PermAddress” as a composite attribute.

The *Reservation* entity contains the serial “ResID” attribute as an identifier and “ResStay” as a composite attribute that denotes the customer’s stay (with the values “CheckInDate” and “CheckOutDate”). Then, the derived attribute “DaysIn” subtracts the “CheckOutDate” from “CheckInDate” in order to generate a bill amount for the customer. The “BillAmt” derived attribute computes the product of “DaysIn” and “Room.RoomPrice” to generate the bill amount. The reason this was made into an associative entity is due to the fact that a reservation requires an ID attribute for itself, and that cannot be represented in a relationship.

The *Room* entity contains the “RoomNum” attribute as an identifier. The attributes RoomType and RoomPrice simply keep track of the room type and its respective price, with both referencing Type.TName and Type.TPrice. Following that entity, the entity *Type* contains TName as its identifier, maintaining the type’s name. Then, the attribute TPrice maintains each type’s price.

The *Makes* relationship does not possess any attributes because its respective attributes were placed in the Reservation associative entity, and its constraints are as follows: a customer can make zero to as many room reservations as they please, but a reservation can only be made by one customer only. Same applies to the *Contains* and *Has* relationships: a room may have zero to many reservations but must have one to many types; a reservation can only apply to one room per instance and a type can only be applied to a specific room per instance.