**Justification for normalized relational schemas**

For conversion between our ER model and our relational model, first, we simply converted every entity in the ER model into its own relation. Next, we did the same thing with all of our relationships. We decided not to include any of our relationships inside of other tables, as otherwise we would have many null values. Overall, we simply followed the conversion algorithm discussed in class.

Below, we show that every table is in at least BCNF,

R1) Relation 1 contains our host information. Each host is identified by their host ID number. All other attributes are dependent on host ID, therefore our left-hand side of our functional dependency (hostID → hostName, hostPhoneNo, hostEmail, hostBankName, hostBankAccountNo) is a key (super key), satisfying the Boyce-Codd Normal Form.

R2) Relation 2 contains our house information. Each house is identified by their host ID number and house ID number jointly. All other attributes are dependent on these two numbers, therefore our left-hand side of our functional dependency (hostID, houseID → houseAddress, houseRate, maxOccupancy) is a key (super key), satisfying the Boyce-Codd Normal Form.

R3) Relation 3 contains all the amenities information to a specific house. Each house is identified by the hostID and houseID number. The rest of the attributes in the relation are all dependent on the hostID and houseID number, therefore the left-hand side of our functional dependency (hostID, houseID → wifi, parking, kitchen, pet\_friendly, washer, dryer) is a key (super key) which satisfies the Boyce-Codd Normal Form.

R4) Relation 4 contains our guest information. Each guest is identified by their guest ID number. All other attributes are dependent on this number, therefore our left-hand side of our functional dependency (guestID → guestName, guestPhoneNo, guestEmail) is a key (super key), satisfying the Boyce-Codd Normal Form.

R5) Relation 5 contains all the pertinent to a reservation. All the attributes in this relation are dependent on the reservation ID (reservationID), therefore the left-hand side of the functional dependency (reservationID → numberGuests, checkInDate, checkOutDate, isCanceled, billDate, billAmount, isPaid) is a key (super key) which satisfies the Boyce-Codd Normal Form.

R6) Relation 6 contains review information. Each review is identified by their review ID number. All other attributes are dependent on this number, therefore our left-hand side of our functional dependency (reviewID → stars, comments) is a key (super key), satisfying the Boyce-Codd Normal Form.

R7) Relation 7 contains payment information. Each payment is identified by their payment ID. All other attributes are dependent on this number, therefore our left-hand side of our functional dependency (paymentID → paymentMethod, CCNumber, nameOnCard, ccExpDate) is a key (super key), satisfying the Boyce-Codd Normal Form.

R8) Relation 8 contains all the information that links all the above relations together. The rerservationID number can be used to uniquely identify every attribute of that tuple. Therefore, the left-hand side of our functional dependency (reservationID → hostID, houseID, guestID, reviewID, paymentID) is a key (super key), which satisfies Boyce-Codd Normal Form.