Question 3.

3(A)

a.

Minimal Basis:

RegNo → Make
RegNo → Model
RegNo → Commission
RegNo → SalesPerson
BuyerName → Address

The key for this relation is {RegNo, BuyerName}.

b.

After doing the minimal basis and determining the key we are left with the following FD's:
FD1. RegNo → Make, Model, Commission, SalesPerson
FD2. Buyername → Address
Unfortunately this is not in BCNF. For this to be in BCNF both FD1 and FD2 need to be superkeys, in this instance neither is a Superkey as RegNo on it's own cannot determine Address or BuyerName, and BuyerName alone can only determine Address.

c.

Constructing Relations:

R1 (RegNo, Make)
R2 (RegNo, Model)
R3 (RegNo, Commission)
R4 (RegNo, SalesPerson)
R5 (BuyerName, Address)

Combining Relations:

CarSales1 (<u>RegNo</u>, Make, Model, Commission, SalesPerson)
CarSales2 (<u>BuyerName</u>, Address)
CarSales3 (<u>RegNo</u>*, <u>BuyerName</u>*)

(B)

To prove this decomposition is incorrect we will complete the 3NF decomposition.

Minimal Basis:

Make, Model → Engine_Size
Registration_No → Make
Registration_No → Colour

Registration_No → Model
Engine_Size → Tow_Load

Using inference rules we can determine that {Registration_No} is the key as it can determine all other values.

Constructing Relations:
R1 (Make, Model, Engine_Size)
R2 (Registration_No, Make)
R3 (Registration_No, Colour)
R4 (Registration_No, Model)
R5 (Engine_Size, Tow_Load)

Combining Relations:
CAR_DETAILS1 (Registration_No, Make, Model, Colour, Engine_Size*)
CAR_DETAILS2 (Engine_Size, Tow_Load)
At this point we have ended with the same decomposition as specified however there is an issue with an FD contained within CAR_DETAILS1 as follows:
Make, Model → Engine_Size is not a valid key and as such this fails BCNF.
To correct this we can combine the relations as follows:
CAR_DETAILS1 (Registration_No, Colour, Make, Model)
CAR_DETAILS2 (Engine_Size, Tow_Load)
CAR_DETAILS3 (Make, Model*, Engine_Size*)