2.

What is the primary key for T?

$\{A, B\} = \{A, B, C, I\}$

$\{B, D\} = \{B, D, E, F\}$

$\{A, D\} = \{A, D, G, H, I, J\}$

So then, $\{A, B, D\} = \{A, B, C, D, E, F, G, H, I, J\}$

<u>Primary Key:</u> $\{A, B, D\}$

Decompose T into 2NF tables.

Primary key is included in the set of tables as R_1, the rest are made so that there are no partial dependencies:

$R_1 = \{\underline{A}, \underline{B}, \underline{D}\}$

$R_2 = \{\underline{A}, \underline{B}, C\}$

$R_3 = \{\underline{B}, \underline{D}, E, F\}$

$R_4 = \{\underline{A}, \underline{D}, G, H, J\}$

$R_5 = \{\underline{A}, I\}$

Decompose T into 3NF tables.

In order to get rid of the transitive dependency R_4 was split into two tables, the new R_4 and the new R_5. The rest of the tables stay the same.

$R_1 = \{\underline{A}, \underline{B}, \underline{D}\}$

$R_2 = \{\underline{A}, \underline{B}, C\}$

$R_3 = \{\underline{B}, \underline{D}, E, F\}$

$R_4 = \{\underline{A}, \underline{D}, G, H\}$

$R_5 = \{\underline{H}, J\}$

$R_6 = \{\underline{A}, I\}$

3.

Based on the primary key, is this table in 1NF, 2NF, or 3NF? Why?

This table is in 1NF because there are no repeating group attributes or atomic values. It cannot be said to be in 2NF because there is a partial dependency. The primary key consists of both Car# AND Salesperson#; however, CommissionPercent is reliant only on Salesperson#, and not on Car# as well.

The table is in 1NF.

How would you successively normalize it completely?

To get it to normalize to 2NF, first you would get rid of the partial dependency of CommisionPercent:

CAR_SALE{<u>Car#,</u>DateSold, <u>Salesperson#</u>, DiscountAmt}

SALE_COMMISSION{<u>Salesperson#</u>, CommissionPercent}

To completely normalize it to 3NF, you must get rid of the transitive dependency of DiscountAmt:

CAR_SALE{<u>Car#,</u>DateSold, <u>Salesperson#</u>}

SALE_COMMISSION{<u>Salesperson#</u>, CommissionPercent}

DISCOUNT{<u>DateSold</u>, DiscountAmt}

4.

What normal form is the table in? Explain your answer.

The primary key of the table is {BookTitle, AuthorName}. Using this we can see that the table is in 1NF because there are no repeating group attributes or atomic values. Also using the primary key, we can see that the table cannot be in 2NF due to the partial dependency of Publisher, BookType, and AuthorAffil. All three of these rely on either BookTitle OR AuthorName, but none rely on both parts of the primary key.

This table is in 1NF.

Apply normalization until you cannot decompose the tables further. State the reasons behind each decomposition.

To get to 2NF, the multiple partial dependencies must be broken up. BOOK is split into three new tables: BOOK1, BOOK2, and BOOK3

BOOK1{<u>BookTitle</u>, <u>AuthorName</u>)

BOOK2{<u>BookTitle</u>, BookType, ListPrice, Publisher}

BOOK3{<u>AuthorName</u>, AuthorAffil}

There is still ListPrice's transitive dependency on BookType that must be dealt with so that the tables can be normalized to 3NF. To do this BOOK2 must be split into the new BOOK2 and the new BOOK3

BOOK1{<u>BookTitle</u>, <u>AuthorName</u>)

BOOK2{<u>BookTitle</u>, BookType, Publisher}

BOOK3{<u>BookType</u>, ListPrice}

BOOK4{<u>AuthorName</u>, AuthorAffil}