Q1. Consider another relation Store = (order, product, customer, address, qty, unitprice) with a set of functional dependencies F= {order →{customer, address}, customer →address, product →unitprice, {order, product} →qty}.

Solution: As the schema is given, the relation is in 1NF.

The primary key is given as {order, product}. Here, the key attributes are: order and product.

In order →{customer, address}, the customer and address attributes are partially dependent on the key. Thus, this relation is not in 2NF. After creating a separate relation R1= (order, customer, address) with functional dependencies set F1={order→{customer, address}, customer→ address}. The original relation becomes R = (order, product, qty, unitprice) with set of FDs F={product→unitprice, {order, product}→qty}.

Still, the relation R is not present in 2NF because unitprice is partially dependent on the key. Thus, a new relation R2 = (product, unitprice) with set of FDs F2 = {product→unitprice} will be created. The relation R is changed to R=(order, product, qty) with set of functional dependencies F={{order, product}→qty}.

Now, all the relations are present in 2NF.

In relation R1, the address attribute is transitively dependent on the key as order→customer and customer→address. Thus, R1 is not in 3NF. At the time of converting the relation to 3NF, create a new relation R3= (customer, address) with the set of FDs F3 = {customer→address}. Now, the parent relation is R1 = (order, customer) with set of FDs F1 = {order→customer}.

Thus, after the conversion into 3NF, the relations are:

* R(order, product, qty) with FD {{order, product}→qty}
* R1=(order, customer) with FD{order→customer}
* R2=(product, unitprice) with FD {product→unitprice}
* R3=(customer, address) with FD {customer→address}

Q2. Consider a relation R = (A, B, C, D, E) with a set of functional dependencies F={A→{B, E}, C→D}. Decompose the relation to BCNF.

Solution: As the schema is given, the relation is in 1NF.

{A}+={A, B, E}

{B}+={B}

{C}+={C, D}

{D}+={D}

{E}+={E}

{A, B}+={A, B, E}

{A, C}+={A, B, C, D, E}

{A, D}+={A, B, D, E}

{A, E}+={A, B, E}

{B, C}+={B, C,D}

{B, D}+={B, D}

{B, E}+={B, E}

{C, D}+={C, D}

{C, E}+={C, D, E}

{D, E}+={D, E}

The candidate key of this relation is: {A, C}. As there is only one candidate key present, the primary key is: {A, C}. The key attributes are: A and C.

In A→{B, E}, the attributes B and E are partially dependent on the key. Thus, this relation is not present in 2NF. To convert the relation into 2NF, construct a new relation R1=(A, B, E) with the set of functional dependencies F1={ A→{B, E}}. The original relation is R= (A, C, D) with the set of FDs F = {{A, C}→D, C→D}.

Still the relation R is not in 2NF as in C→D, D is partially dependent on the key. Thus, construct a new relation R2 = (C, D) with the set of FDs F2={C→D}. The original relation is now R=(A,C) with the set of FDs F={{A, C}→{A, C}}.

As there is no non-key attribute transitively dependent on the key, all the relations are in 3NF.

Similarly, all the determinants are the keys. Thus, the relations are present in BCNF. After BCNF, the relations are:

* R=(A, C) with FD {{A, C}→{A, C}}
* R1=(A, B, E) with FD { A→{B, E}}
* R2 = (C, D) with FD {C→D}