**Relational Schema:**

**Entity: User**
**Attributes:**
User_Id(PK)
Name(Composite Attribute)
Phone
Email
Password
Address(Composite Attribute)

**1NF:-** This relation doesn't satisfy the 1NF condition because of the presence of composite attributes.So,to bring this relation into 1NF we have to insert sub attributes instead of main attributes.

**Attributes:**
User_id(PK)
First_Name
Last_Name
Phone
Email
Password
Pincode
State
Country

**Functional Dependencies:**

- User_id → First_Name
- User_id → Last_Name
- User_id → Password
- User_id → Phone
- User_id → Email
- User_id → Pincode
- User_id → State
- User_id → Country
- Pincode → State
- State → Country

**2NF:** As the determinant of the above functional dependencies is single valued attribute.As no proper subset of candidate key exists and no chance of existence of partial dependency,the above relation is in 2NF.

**3NF:** In the above Functional dependencies, Pincode and state are nonprime attributes and they are determining another non-prime attribute country.So, transitive dependency exist .Therefore this table is not in 3NF.
To bring this table into 3NF we should do lossless decomposition.

**Decomposition into tables:**
**1) USER** - user_id, First_Name, Last_Name, email, Password, Pincode
**2) PIN** - Pincode, State

**3) STATE** - State, Country

**BCNF :-** In all the above modified tables , only the superkeys are determining all other attributes.
Hence, we can say that the table is in BCNF.

**Entity: Category**
**Attributes:**
Category_Name
Category_id(PK)

**Functional Dependencies:**

- Category_id→Category_Name

This relation satisfies the 1NF condition as it ensures that each column contains atomic values with no repeating groups. Furthermore, it has no partial dependencies because all columns are functionally dependent on the primary key. Therefore, the relation satisfies the 2NF condition.
All functional dependencies are from the candidate key (prime attribute) to non-prime attributes, eliminating partial dependencies. Consequently, the table is also in 3NF since there are no transitive dependencies between non-prime attributes.Finally, all functional dependencies are from the superkey, Payment_id, to all other attributes, confirming that this table is in BCNF.

**Entity: Product**
**Attributes:**
Product_id(PK)
Category_id(FK)
Product_Name
Description
Price

**Functional Dependencies:**

- Product_id →Category_id
- Product_id→Product_Name
- Product_id→Description
- Product_id→ Price
- Product_Name→Price
- Product_Name→Description

This relation satisfies the 1NF condition as it ensures that each column contains atomic values with no repeating groups. Furthermore, it has no partial dependencies because all columns are functionally dependent on the primary key. Therefore, the relation satisfies the 2NF condition.

In the above Functional dependencies Product_Name is nonprime attributes and it is determining another non-prime attribute country. So,transitive dependency exist .Therefore this table is not in 3NF.
To bring this table into 3NF we should do lossless decomposition.

**Decomposition into tables:**

1)**Product**:Product_id,Category_id,Product_Name
2)**Price:**Product_Name,Description,Price

In the above tables, as all functional dependencies are from the superkeys to all other attributes, confirming that this tables are in BCNF.

**Entity: Cart**
**Attributes:**
Cart_id(PK)
User_id(FK)
Product_id(PK)
Quantity

**Functional dependencies:**

- Cart_id→User_id
- Product_id,Cart_id→Quantity

**1NF:**This relation satisfies the 1NF condition as it ensures that each column contains atomic values with no repeating groups.

**2NF:**In this table (Cart_id,Product_id) becomes the primary key. User_id attribute partially depends on the primary key i.e., on the Cart_id prime attribute.So there is Partially Dependencies.To bring this table into 2NF we should do lossless decomposition.

**Decomposition into tables:**

**Cart:**Cart_id,User_id
**Product_Cart:**Cart_id, Product_id,Quantity

**3NF:**The table is in 3NF since there are no transitive dependencies between non-prime attributes.

**BCNF:**In the above tables, as all functional dependencies are from the superkeys to all other attributes, confirming that this tables are in BCNF.

**Entity: Orders**

**Attributes:**

Order_id(PK)
Cart_id(FK)
Status
TotalPrice
OrderDate

**Functional dependencies:**

- Order_id→Cart_id
- Order_id→Status
- Order_id→TotalPrice
- Order_id→OrderDate

This relation satisfies the 1NF condition as it ensures that each column contains atomic values with no repeating groups. Furthermore, it has no partial dependencies because all columns are functionally dependent on the primary key. Therefore, the relation satisfies the 2NF condition.
All functional dependencies are from the candidate key (prime attribute) to non-prime attributes, eliminating partial dependencies. Consequently, the table is also in 3NF since there are no transitive dependencies between non-prime attributes.Finally, all functional dependencies are from the superkey, Payment_id, to all other attributes, confirming that this table is in BCNF.

**Entity: Payment**
**Attributes:**
Payment_id(PK)
Order_id(FK)
Amount
PaymentDate
PaymentMethod

Note:Here amount includes TotalPrice from orders plus other charges including like delivery charges.
**Functional Dependencies:**

- Payment_id→Amount
- Payment_id→Order_id
- Payment_id→PaymentDate