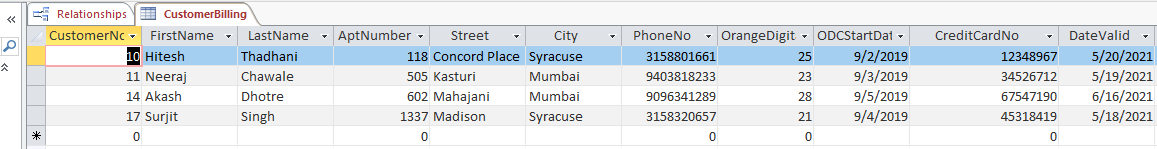
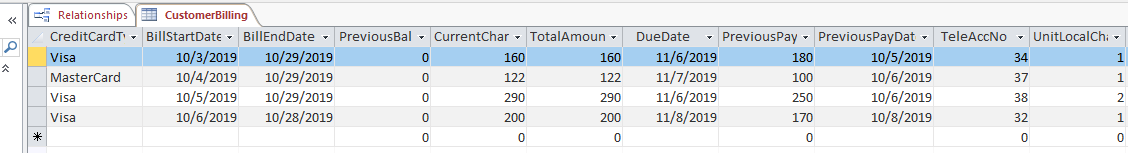
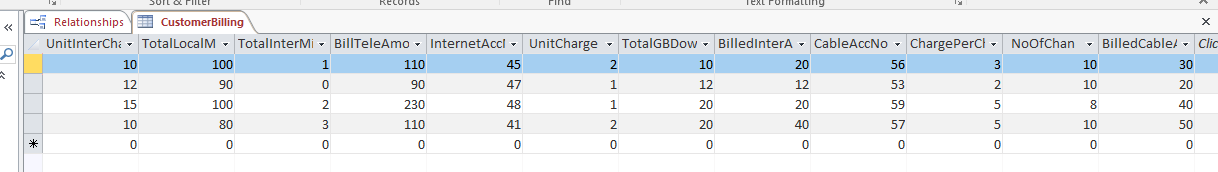
**Normalization of Digital Bill Summary for Orange Digital Company**

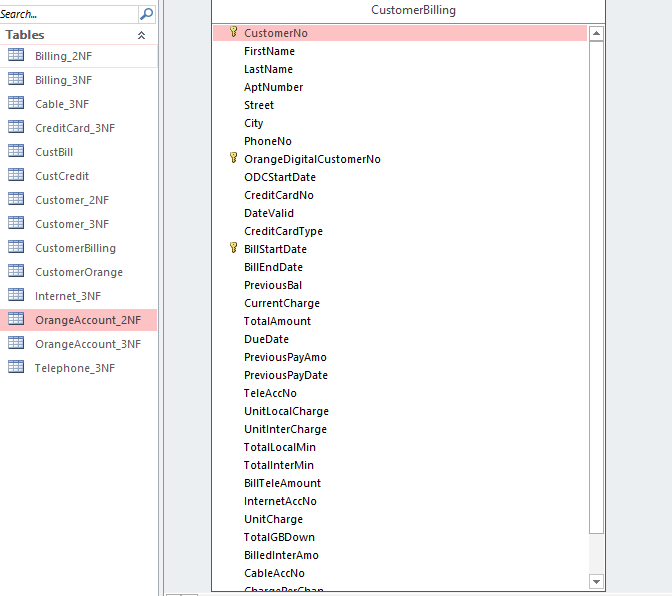
**First Normal Form (1NF):**

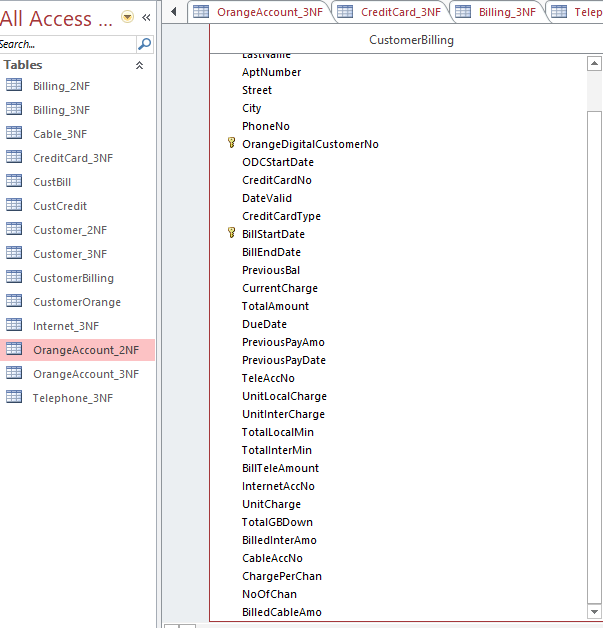






Customer Table in 1NF Form





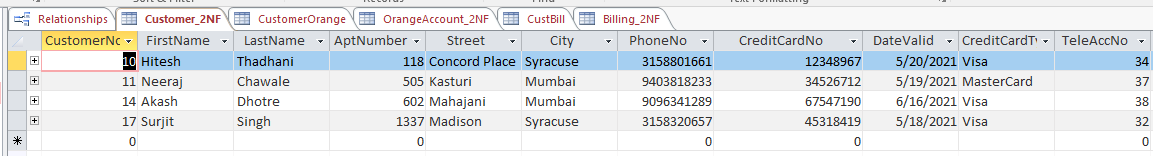
**Relationship in 1NF Form in MS Access**

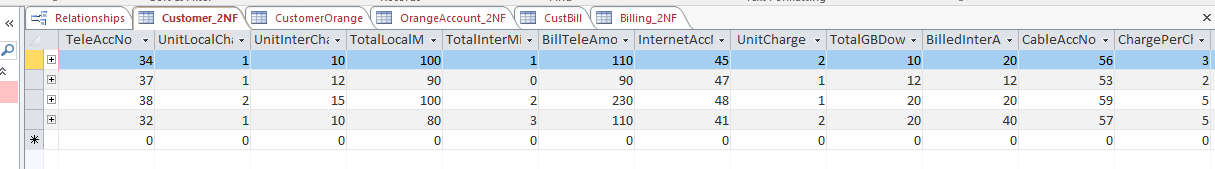
**Conversion from 1NF to 2NF Form:**

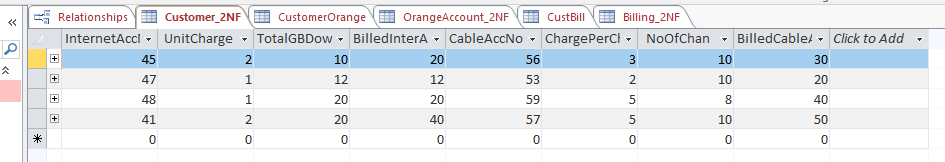
In 1NF form we have all the relations in atomic values no repeating or multivalued attributes. To convert from 1NF to 2NF Form, we have to identify the Functional Dependencies and resolve them to be in 2NF Form.

**Customer Table 1NF to 2NF:**

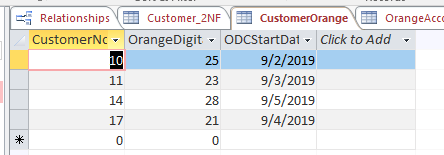
Here Customer ID is composite PK on which FirstName, LastName, AptNo, Street, City, PhoneNo depend in other words this Customer ID has partial dependencies with these customer attributes. Along with that Each Credit Card and the card details would form a partial dependency on Customer ID. Each Customer has 1 and only 1 Telephone, Internet and Cable Account number with their respective attributes identified by Customer ID. So Customer ID is has partial dependency with customer details, credit card details and Usage details for all 3 services. Hence Customer Table has all these fields in 2NF Form.



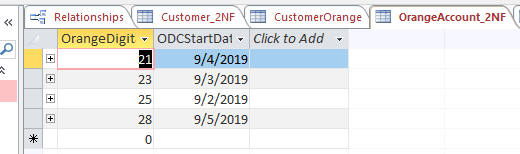




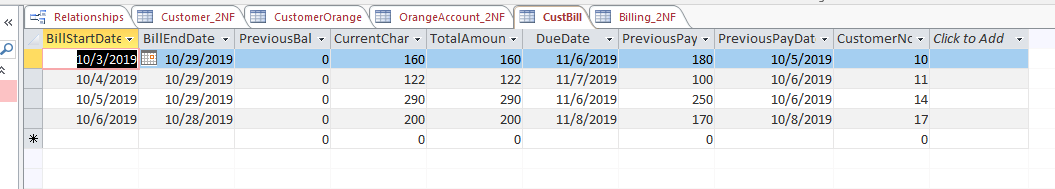
Customer table in 2NF Form



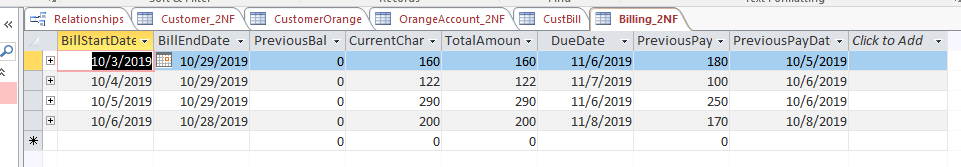
CustomerOrange Table(Associative Entity) Table



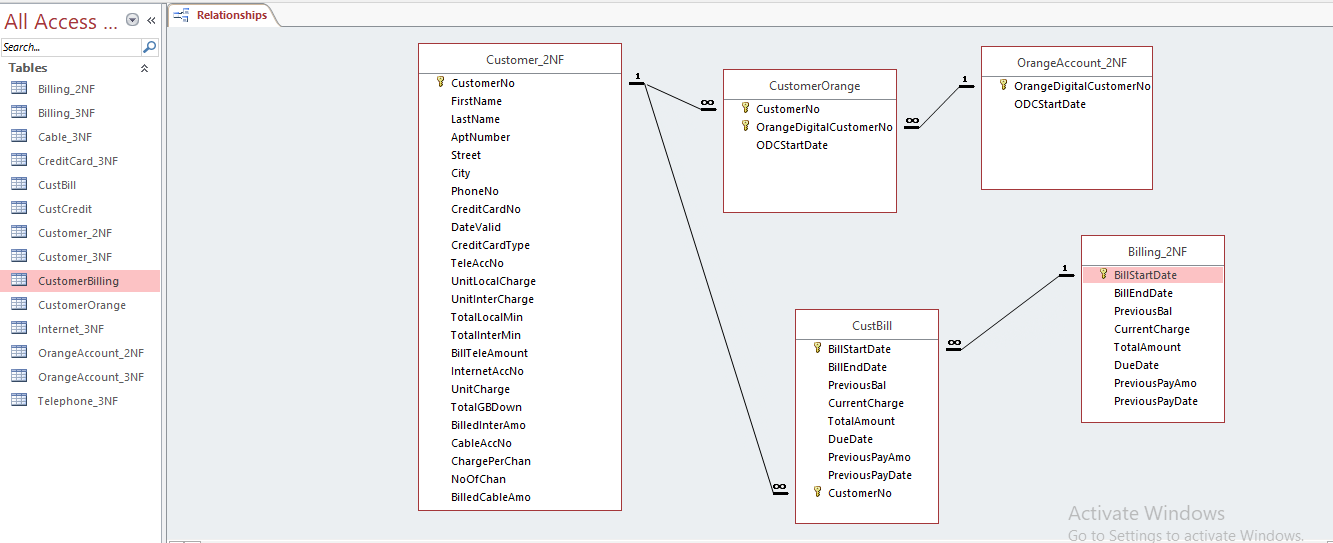
OrangeAccount in 2NF Form where StartDate is dependent on Composite PK OrangeDigitalAccountNo.



CustBill Table(Associative Entity)



Billing Table in 2NF Form where all the values are dependent on Composite PK BillStartDate.



**Relationship in 2NF Form in MS Access**

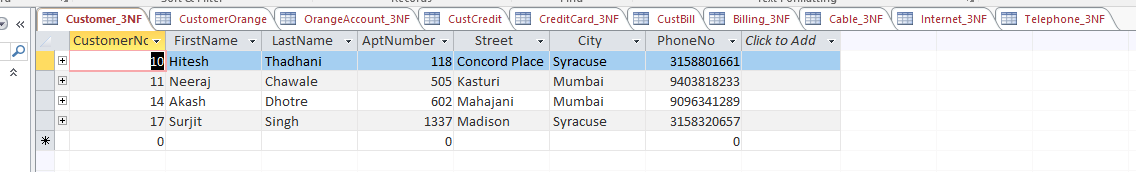
Here 2 Associative Entities CustomerOrange and CustBill are used for Many-to-Many Relationship between Customer & OrangeAccount, Billing respectively. Here I have used Composite PK FK in the Associative Entity to create 2 1:M relationships. Along with Removing Partial Dependencies to make it into 2NF Form.

**Conversion from 2NF to 3NF Form:**

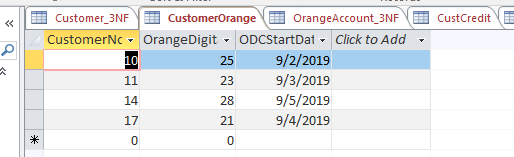
In 2NF form we have all the relations in atomic values no repeating or multivalued attributes along with functional dependencies resolved. To Convert from 2NF to 3NF Form, we have to identify the Transitive Dependencies by finding one non-key attribute on which other non-key attributes depend and resolve them to be in 3NF Form where each attribute is dependent on PK only.

**Customer Table in 3NF Form:**

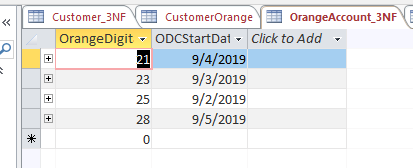
Here Only Customer details are included including FirstName, LastName, AptNo, Street, City, PhoneNo. All the other details which exists in 2NF are have transitive dependencies. Credit Card Number becomes PK for Credit Card details so all the attributes of credit card are only dependent on Card No. Usage details are also removed based on TelephoneAccountNo, Internet Acc No and Cable Acc No which have 1 to 1 relationship with Customer ID.



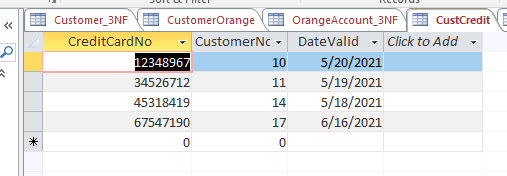
Customer Table in 3NF Form where all the attributes depend on PK Customer ID only.



CustomerOrange Table(Associative Entity)

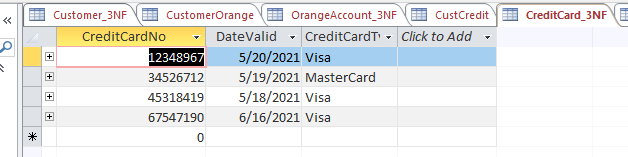


OrangeAccount Table in 3NF Form where StartDate is dependent only on Orange Account No.

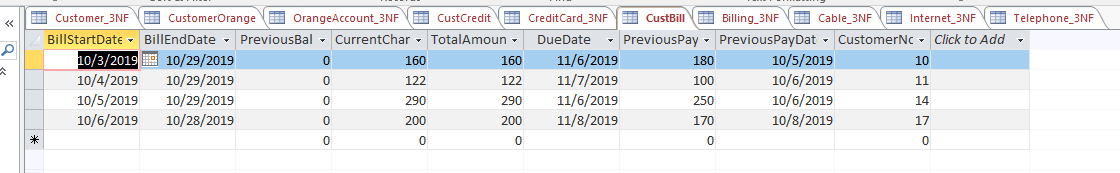


CustCredit Table(Associative Entity)

**Credit Card 3NF:** Credit Card Number is a non-key attribute which has transitive dependency on Card Validity, Type of the Credit Card. So Credit Card No is the Primary Key here on which other details are based like Card validity, Type. Credit Card has Many-to-Many relationships between Customer ID. 1 Customer can have 1 or more CC and 1 or more CC can belong to 1 customer. So we need Associative entity to break this M:N Relationship into 2 1:M relationship so a new Associative entity is created CustCredit using **Composite Key** as PK,FK so it doesn’t take any NULL Values in foreign keys.

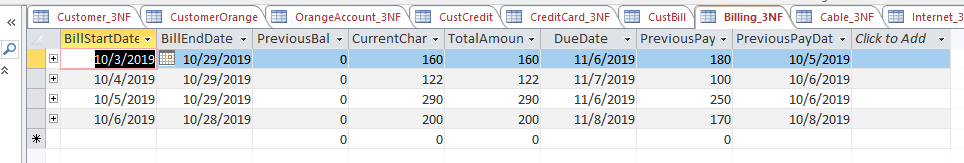


CreditCard Table in 3NF derived after resolving transitive dependencies.



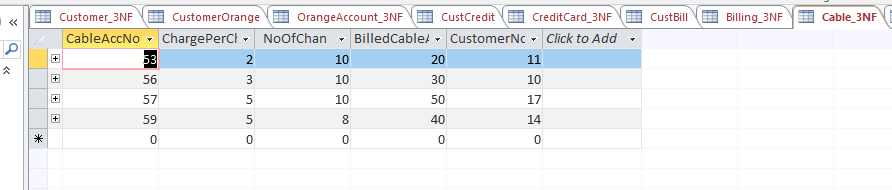
CustBill Table (Associative Entity)

**Billing in 3NF Form:** Billing has all the attributes which are derived keeping BillStartDate PK.



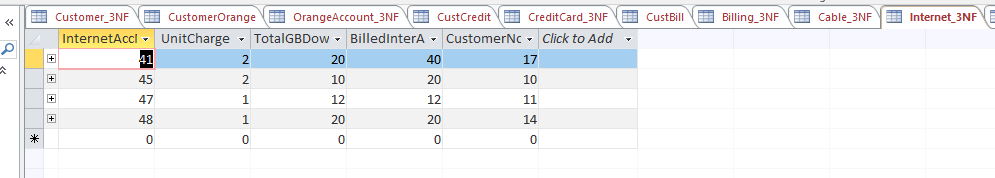
Billing Table in 3NF Form

**CableBill 3NF Form:** Cable Bill was part of Customer Table in 2NF Form and CableAccNo has Transitive Dependencies with ChargesperChannel, NoOf Channels, BilledCableAmount. CustomerID is included as it is 1:1 Relationship between them to have a common column to connect both the tables **as a Foreign Key**. So CableAccNo is a PK where all the other non-key attributes depend only on the PK resolving Transitive Dependency and creating a different table for it.



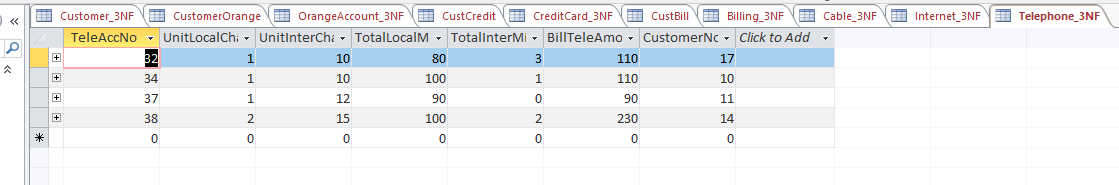
CableBill Table in 3NF Form

**InternetBill 3NF Form:** Internet Bill was part of Customer Table in 2NF Form and InternetAccNo has Transitive Dependencies with UnitCharge, TotalGBDownloaded, BilledInternetAmount . CustomerID is included as it is 1:1 Relationship between them to have a common column to connect both the tables **as a Foreign Key**. So InternetAccNo is a PK where all the other non-key attributes depend only on the PK resolving Transitive Dependency and creating a different table for it.

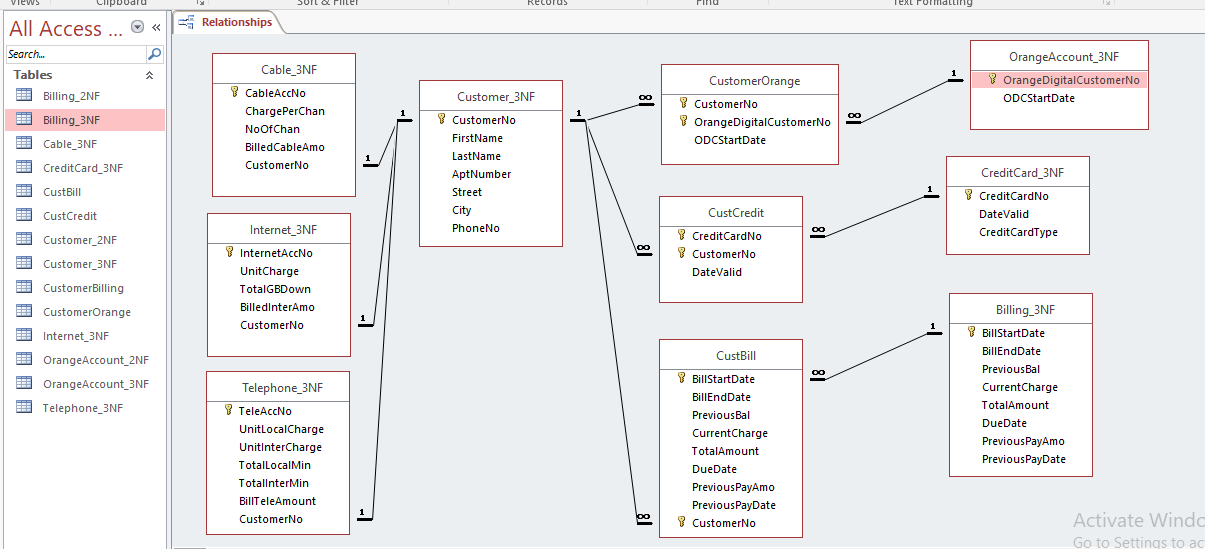


InternetBill Table in 3NF Form

**TelephoneBill 3NF Form:** Internet Bill was part of Customer Table in 2NF Form and TelephoneAccNo has Transitive Dependencies with UnitLocalCharge, UnitInternationalCharge, TotalLocalMinutes, TotalInternationalMinutes, BilledTelephoneAmount. CustomerID is included as it is 1:1 Relationship between them to have a common column to connect both the tables **as a Foreign Key**. So TelephoneAccNo is a PK where all the other non-key attributes depend only on the PK resolving Transitive Dependency and creating a different table for it.



TelephoneBill Table in 3NF Form



**Relationship in 3NF Form in MS Access**

**Foreign Key Constraints between TelephoneBill , InternetBill, CableBill and Customer Table**:

TelephoneBill Table

Customer Table

CustomerID FirstName …….

TeleAccNo UnitLoChargeUnitInCharg …… CustomerID

InternetBill Table

InternetAccNo UnitCharge TotalGBDownl …… CustomerID

CustomerID FirstName …….

Customer Table

CableBill Table:

Customer Table

CableAccNo ChargePerChan NoofChan …… CustomerID

CustomerID FirstName …….