**B. Description**

1. In this diagram the entities are Artworks, Artists, Customer and Group.
2. Artist paints artworks so they are connected by the relationship ‘Paints’.
3. Artworks are classified in groups. They are connected by the relationship ‘Classify’.
4. Customer buys artworks so they are connected by the relationship ‘Buy’. ‘Buy’ has two attributes ‘Date’ and ‘Txn\_ID’.
5. A customer prefers some artists. So customer and artists are connected by the relationship ‘Prefers’.
6. Customer also likes some artwork groups so, group and customer is connected by the relationship ‘Like’.

**C. Attributes**

Artwork: {Title, Type, Year\_of\_Made, Price}

Artists: {Name , Style\_of\_Art, Age, Addrress}

Customer: {ID, C\_Name, C\_Address, Total\_spent}

Group: {GID ,G\_Name}

Buy: {Txn\_ID , Date}

**4.Relationships**

a. Artists-Artworks (1:N) comprises.

b. Customer-Artists (N:M) comprises.

c. Customer-Artworks (1:N) comprises.

d. Customer-Group (N:M) comprises.

e. Artworks-group (M:N) holds.

**5. Relational Schema**

**Artworks:**

Artworks

<Title>

Type of art : <Type>

Year of art was made : <Year\_of\_Made>

Price of artwork : <Price>

Artist name : <Name> (Forgien key from ‘Artists’ table)

Customer id : <ID> (Forgien key from ‘Customer’ table)

Date of sold : <Date>

Transaction id : <Txn\_ID>

**Artists:**

Artists

<Name>

Artist age : <Age>

Artist address : <Address>

Style of art : <Style\_of\_Art>

**Customer:**

Customer

<ID>

Customer name : <C\_Name>

Customer address : <C\_Address>

Amount spent in gallery : <Total\_Spent>

**Group:**

Group

<GID>

Group name : <G\_Name>

**Prefers:**

Prefers

Artist name : <Name> (Forgien key from ‘Artists’ table)

Customer id : <ID> (Forgien key from ‘Customer’ table)

**Like:**

Like

Group id : <GID> (Forgien key from ‘Group’ table)

Customer id : <ID> (Forgien key from ‘Customer’ table)

**Classify:**

Classify

Group id :<GID> (Forgien key from ‘Group’ table)

Artwork title : <Title> (Forgien key from ‘Artworks’ table)

**6. Normalization of Relational Schema**

**Artwork {Title, Type, Year\_of\_Made, Price , Name, ID, Date, Txn\_ID}**

FD->

Title-> {Type, Year\_of\_Made, Price, Name, ID, Date, Txn\_ID}

Txn\_ID-> {Date, ID}

Candidate key -> Title

Here no attributes are multi valued So, table is in **1NF**.

Here Title is the candidate key and Txn\_ID is non prime attribute ,so there are no Partial dependency Present. so, table is in **2NF**.

Here Txn\_ID is a non prime attribute and Date , ID is depend on that, so table is **not** in **3NF.**

Now the decompose table will be Artwork{Title, Type, Year\_of\_Made, Price, Name,Txn\_ID} And Another table will be Transaction{ Txn\_ID, Date, ID}.

Now from the new tables Artworks candidate key will remain same “Title” and Transaction table’s candidate key will be “Txn\_ID”. Now both table Artworks and Transaction is in **3NF**.

In Artworks table Title is the candidate key so the table is in **BCNF,** and in Transaction table candidate key is Txn\_ID so, table is also in **BCNF**.

**Artists{Name, Age, Address, Style\_of\_Art}**

FD->

Name->Age, Address, Style\_of\_Art

Candidate key -> Name

Here no Attributes are Multi Valued So, Table is in 1NF.

Here all attributes dependent on “Name” that is candidate key no partial dependencies are present so, table is in 2NF.

Here all attributes dependent on “Name” that is candidate key no Transitive dependencies are present so, table is in 3NF.

Here all attributes dependent on candidate so, table also in BCNF.

**CUSTOMER {ID, C\_Name, C\_Address, Total\_Spent}**

FD->

ID-> C\_Name, C\_address, Total\_Spent

Candidate Key-> ID

Here no Attributes are Multi Valued So, Table is in 1NF.

Here all attributes dependent on “ID” that is candidate key no partial dependencies are present so, table is in 2NF.

Here all attributes dependent on “ID” that is candidate key no Transitive dependencies are present so, table is in 3NF.

Here all attributes dependent on candidate so, table also in BCNF

**GROUP{GID, G\_Name}**

FD:

GID-> G\_Name

Candidate Key-> GID

Here no Attributes are Multi Valued So, Table is in 1NF.

Here all attributes dependent on “GID” that is candidate key no partial dependencies are present so, table is in 2NF.

Here all attributes dependent on “GID” that is candidate key no Transitive dependencies are present so, table is in 3NF.

Here all attributes dependent on candidate so, table also in BCNF

**Prefers {Name, ID}**

Candidate Key-> {Name, ID}

Here no Attributes are Multi Valued So, Table is in 1NF.

Here no attributes are dependent on any other attribute, all the attributes together form a candidate key so, table is in 2NF, 3NF, BCNF.

**Like {GID, ID}**

Candidate Key-> {GID,ID}

Here no Attributes are Multi Valued So, Table is in 1NF.

Here no attributes are dependent on any other attribute, all the attributes together form a candidate key so, table is in 2NF, 3NF, BCNF.

**Classify{GID, Title}**

Candidate Key-> {GID, Title}

Here no Attributes are Multi Valued So, Table is in 1NF.

Here no attributes are dependent on any other attribute, all the attributes together form a candidate key so, table is in 2NF, 3NF, BCNF.