**Exercise 1 – Normalization**

**Antique dealer database – Solution**

**Lets look at the attributes and the data:**

**The unnormalized attributes:**

Item No.,

Description,

Price (£),

Order\_Number,

Customer\_Number,

Customer\_Name,

Customer\_Address,

Customer\_Tel\_No,

Date\_of\_Purchase,

Delivered,

Payment\_Card,

Card\_Type,

Expiry\_Date

The above data is in an unnormalized form.

Conditions for 1NF:

1. No repeating groups.
2. Each cell must contain atomic value.
3. Each record must be unique (no duplicate rows) – every record should be identified by a unique identifier.

* If you look at it closely, you can see that we take a combination of PKs like ItemNo, Order\_Number and Customer\_Number to identify each record uniquely.
* If we do the above, this will automatically identify Item (Item No, Description, Price (£))as a repeating group that needs to be separated from the rest of the data as some of the rows of the original data do not contain any other values.
* Performing the above steps ensures we have atomic values in each record as we end up with the following tables which are now in 1NF:

**ITEM** (**Item\_No**., Description, Price (£))

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Customer\_Name, Customer\_Address, Customer\_Tel\_No, Date\_of\_Purchase, Delivered, Payment\_Card, Card\_Type, Expiry\_Date)

**ITEM** (**Item\_No**., Description, Price (£))

A black background with a black square

Description automatically generated with medium confidence

Note: ItemNo Uniquely identifies the record. ItemNo is a primary key in the ITEM table.

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Customer\_Name, Customer\_Address, Customer\_Tel\_No, Date\_of\_Purchase, Delivered, Payment\_Card, Card\_Type, Expiry\_Date)

A black background with a black square

Description automatically generated with medium confidence

Order\_number, Customer\_number, ItemNo (composite key) identifies each records uniquely.

Now lets look at the 2NF. In 2NF below are the conditions:

1. It must in 1 NF
2. Non-identifying attributes should not partially be dependent on part of the PK.

* The first condition is satisfied because it is in 1NF and we proved it in the above step. Now about the 2nd condition:
* It doesn’t exist in the ITEM table, So no change there. However, in the ORDER DETAILS table where the PK is a combination of three attributes. There is Customer\_Name, Customer\_Address, Customer\_Tel\_No depending on the Customer\_Number only. Therefore we form a new entity CUSTOMER with Customer\_Number being the identifying attribute. So now, the original data will be split into three tables and it satisfies the conditions for 2NF:

**ITEM** (**Item\_No**., Description, Price (£))

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Date\_of\_Purchase, Delivered, Payment\_Card, Card\_Type, Expiry\_Date)

**CUSTOMER (Customer\_Number**, Customer\_Name, Customer\_Address, Customer\_Tel\_No)

**ITEM** (**Item\_No**., Description, Price (£))

A black background with a black square

Description automatically generated with medium confidence

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Date\_of\_Purchase, Delivered, Payment\_Card, Card\_Type, Expiry\_Date)

A black background with a black square

Description automatically generated with medium confidence

**CUSTOMER (Customer\_Number**, Customer\_Name, Customer\_Address, Customer\_Tel\_No)

A black background with a black square

Description automatically generated with medium confidence

Lets look at the 3NF now. To be in 3NF the following conditions must be satisfied:

1. It should be in 2NF
2. There should be any transitive dependencies (i.e. no dependency between non identifying attributes)

If we look into the tables, we can see that ITEM table and CUSTOMER table satisfies both the above conditions. However, ORDER DETAILS satisfies only the first condition and there is a transitive dependency existing in the table – which needs to be removed. There are attributes Card\_Type and Expiry\_Date which depend on Payment\_Card. As this is a case of transitive dependencies, these should be removed by creating a new entity CARD\_DETAILS. Therefore, we end up with four tables which are all in 3NF:

**ITEM** (**Item\_No**., Description, Price (£))

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Date\_of\_Purchase, Delivered, Payment\_card)

**CUSTOMER (Customer\_Number**, Customer\_Name, Customer\_Address, Customer\_Tel\_No)

**CARD\_DETAILS(Payment\_card,** Card\_Type, Expiry\_Date)

**ITEM** (**Item\_No**., Description, Price (£))

A black background with a black square

Description automatically generated with medium confidence

**CUSTOMER (Customer\_Number**, Customer\_Name, Customer\_Address, Customer\_Tel\_No)

A black background with a black square

Description automatically generated with medium confidence

**ORDER\_DETAILS** (**Order\_Number, Customer\_Number, ItemNo**, Date\_of\_Purchase, Delivered, Payment\_card)

A black background with a black square

Description automatically generated with medium confidence

**CARD\_DETAILS(Payment\_card,** Card\_Type, Expiry\_Date)

A black background with a black square

Description automatically generated with medium confidence