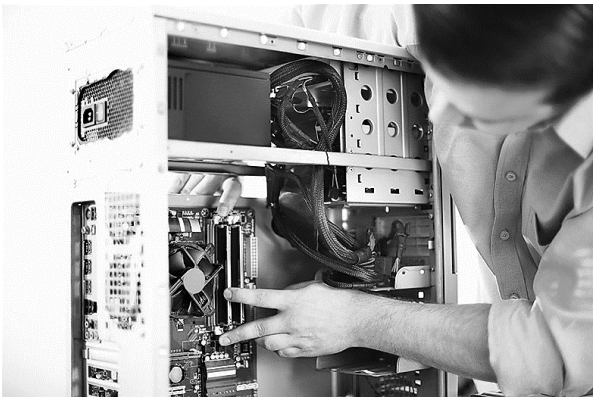
Normalisation Test, SBA Portfolio 2020

Examiner:
Moderator:
Marks: 40
Time: 45 minutes

A local computer shop provides a service to build desktop computers for their customers. The shop provides build options for different needs. For example the “OfficeAdmin” build for every day office use, or the “ProGamer” build for gamers requiring a powerful gaming computer.



https://slickdeals.net/article/news/how-to-build-your-pc/

The computer shop has started to keep track of pc build orders in a database named **PCBuildsDB**. Unfortunately, their database has not been designed very well and they have run into some problems whilst trying to use it.

The data is stored in a table named **tblOrders** with the following fields:

Field	Description
Build	Name of PC build
BuildPrice	Price charged for a single computer build
Processor	Name of CPU
Graphics	Name of Graphics Card/Onboard graphics
RAM	Amount of RAM
Storage	Amount of storage and whether SSD or HDD
Customer	Name of customer
Address	Customer address used for delivery of computer
Phone	Customer phone number
Company/Private	Whether the order is for a company or a private customer
Discount	Companies are given a 5% discount
OrderDate	Date when the order was placed
Quantity	The amount of computers ordered

The computer shop provides the build service to private clients and to businesses. If the order is for a business, then a 5% discount is given.

The database table, **tblOrders**, is printed as appendix A attached.

1 **tblOrders** uses the **Build** and **Customer** fields together as the primary key.

1.1 Define the term primary key.

A field or fields ✔ that uniquely identifies a record ✔

(2)

1.2 What do you call a primary key that is made up of more than one field?

Composite/compound key (1)

1.3 This choice of primary key is very poor. Explain why choosing the **Build** and **Customer** fields as the primary key will not work, in the long term.

Adding a second order for the same build from the same customer will not be possible as the primary key is made up of those two fields.

(2)

2 A badly designed database table, like **tblOrders**, will often contain data dependencies.

2.1 Define a partial dependency.

A field that is dependent on another field that is part of a primary key

(1)

2.2 Define a transitive dependency.

A field that is dependent on a non-key field

(1)

2.3 Identify any partial and transitive dependencies that might be present by drawing a dependency diagram for **tblOrders**.

4 marks for identifying dependencies correctly
2 marks for correct arrow direction and labelling Partial and Transitive

(6)

3 There are some examples of non-atomic data in this database.

3.1 What is non-atomic data?

Data that is not in its simplest form and can be further split

(1)

3.2 Give one example of non-atomic data from the database and explain why it is non-atomic.

Customer name OR Address (You could argue for Processor) + reasonable expl.

(2)

Build	BuildPrice	Processor	Graphics	RAM	Storage	Customer	Address	Phone	Company/ Private	Discount	OrderDate	Quantity

4 One of the advantages of normalisation is that it prevents anomalies from occurring. Explain how each type of anomaly might occur by using examples from **tblOrders**.

Anomaly	Example from tblOrders
Update	Updating customer name/address might need to done in multiple records and an error could occur OR Updating build name or build components might need to done in multiple records and an error could occur
Insert	Adding a new customer requires that they have already placed an order which results in an insert anomaly OR adding a new build requires that it has been ordered by a customer
Delete	Deleting a particular build from the database might result in deleting the details of a customer as well OR deleting a customer could result in the loss of a build

(6)

5 Normalisation also reduces data redundancy.

5.1 Explain how redundant data is different to duplicate data.

Redundant data is needlessly repeated data that results from repeating groups

Duplicate data is when a record has the same value for a field as another record

(2)

5.2 Give an example of redundant data from **tblOrders**.

Any of the build components of repeated builds is redundant (1)

5.3 Give an example of duplicate data from **tblOrders**.

The 500GB HDD for HomeOffice and OfficeAdmin is duplicate but not redundant (1)

6 List the characteristics for each of the following normal forms:

Normal Form	Characteristic
1NF	No repeating groups
	Primary Key identified
2NF	Meets requirements of 1NF
	No partial dependencies
3NF	Meets requirements of 2NF
	No transitive dependencies

(6)

7 Normalise **tblOrders** to 3NF. Use relational notation to write out your answer.

tblCustomers(Customer, Address, Phone, Company/Private) ✔✔

tblBuilds(Build, BuildPrice, Processor, Graphics, RAM, Storage) ✔✔

tblOrders(OrderID, Customer, Build, OrderDate, Quantity) ✔✔

tblDiscount(Company/Private, Discount) ✔✔

CustomerID and BuildID are good answers as well

Primary keys must be underlined

(8)

Appendix A – tblOrders

TOTAL: 40