**Relational Modelling Quiz I**

**Question 1**

Relational modelling takes time and skill. Why do some companies still use relational databases?

* Database Management Systems are mature, reliable software that provides many tools to automate queries and calculations, eliminating human error.

**Question 2**

What is the best approach to design a relational database that is easy to use for its intended purpose?

* We have to ensure we find out exactly what the database is intended to be used for and then design it for this intended use according to the relational principles.

**Question 3**

Suppose that in our model, we have created an attribute called 'address'. It contains the street address, postcode and city of a warehouse location, with a datatype of VARCHAR. What, do you think, is the worst feature of this attribute?

* It is not atomic enough, it would be more useful to have three separate attributes that would make it easier to search addresses and format them according to the context (such as an printing on an envelope).

**Question 4**

Anu is modelling an Australian postcode and at first considers an INTEGER type for the postcode, but then changes it to a CHAR of length 4. Which of the following is true about this choice?

* Anu made a good choice. CHAR of length 4 is the best choice for an Australian postcode.

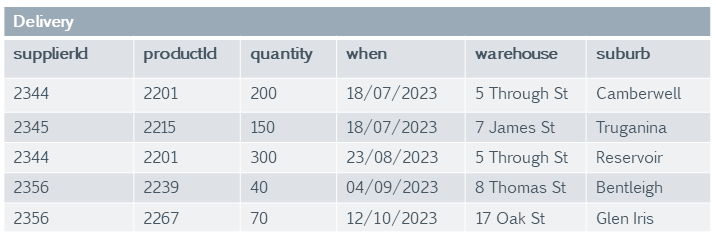
**Question 5**

Anu is continuing work on her design. She decides to add an entity that stores the total daily sales as an aggregate, to save time calculating these values at the end of the month for reporting. She calculates the daily sales values by multiplying product quantity with price (from the sales table), then adding them up for each date and storing them in a table called daily sales. What would be your take on this design?

* It violates relational principles because it introduces redundancy. Even though the numbers are aggregated, they still represent the same information as the raw sales data.

**Question 6**

The Delivery table records all the deliveries we receive from our suppliers. The supplier ID identifies the supplier, and the product ID identifies the item we want to buy. The warehouse address shows where we want the items delivered. The same product always goes to the same warehouse address.

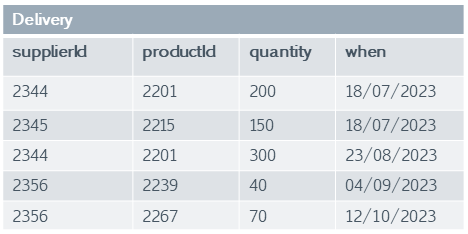


Should we keep the warehouse address in the Delivery table?

* The warehouse address should be shifted to the product table, because the warehouse is product-specific.

**Question 7**

The Delivery table records all the deliveries we receive from our suppliers. The supplier ID identifies the supplier, and the product ID identifies the item we want to buy.

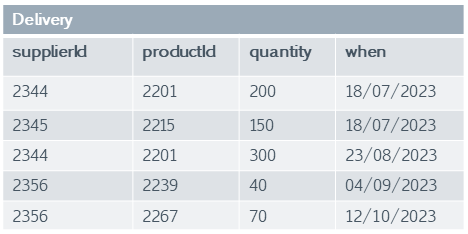


The supplier ID doesn't tell us much about the supplier, so we are considering adding the supplier's name to this table. Do you think this is a good idea?

* No, the supplier's name belongs into a table that is called supplier and has the supplier ID as a primary key.

**Question 8**

The Delivery table records all the deliveries we receive from our suppliers. The supplier ID identifies the supplier, and the product ID identifies the item we want to buy. The 'when' attribute is the delivery date, and the quantity is how many of the product items are being delivered.

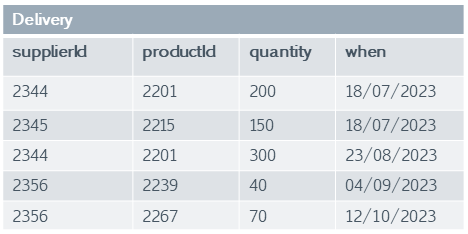


A table needs a primary key to uniquely identify its entities (row data). What would be the best key for this table, assuming we do not add any more attributes?

* The productId, supplierId and 'when' (the delivery date) attributes as a composite key.

**Question 9**

With the delivery table (as shown), we usually assume that each product can potentially be delivered by several suppliers.

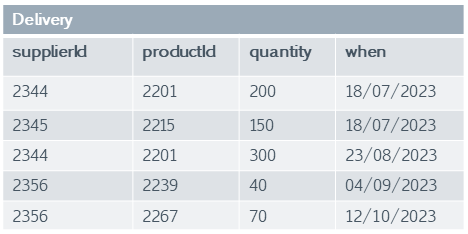


Good business practice would be to order from the supplier with the cheapest price, but switch to another supplier in case of availability issues. Suppose we deal in handmade items that have only one supplier each. How would that affect the Delivery table?

* We would no longer need to include the supplier in the Delivery table, because we would keep a record of suppliers and their products elsewhere in the database.

**Question 10**

We decide that the delivery table (as shown) needs a surrogate key.



Which of the following would be an appropriate choice of surrogate key?

* Creating a deliveryId attribute as an an 'autonumber' (a number that the database manages to make sure it is unique).