

1. PROBLEM DESCRIPTION

The ecommerce company *CaffeineManiacs™ Inc.* needs a new database for recording and handling their data. In order to do this, they have made a specification about their services, providers and clients, as briefed in the following.

The company has a catalog of products, which are identified by their name (unique) and characterized by the coffee (that is, the species of coffee plant); the varietal (a variety of that coffee; there are many, such as Blue Mountain, Colombia, Bourbon, etc.); the origin (the country where that coffee comes from); the roasting type, which can be 'natural', 'high-roast' (*torrefacto*), or 'mixture'; whether it is decaf (decaffeinated) or not; and the marketing formats available for each specific product: raw grain, roasted beans, ground, freeze-dried (soluble), in capsules, or prepared. Any product can be commercialized in several formats, and each format in turn can be packaged differing amounts (weight or volume, depending on the format).

Any available reference is, therefore, an item identified by a bar-code, a packaging description (amount of product), and its retail prize. All references also have a certain number of available items for sale in our warehouses (stock). Whenever a purchase is made, the available units of the involved references should be updated. When the available amount is less than a certain minimum threshold, a new replacement order will be automatically registered. The minimum stock is, in general, different for each reference (by default, 5 units). An automatic process is run daily (usually by night) for setting the existent 'draft' replacement orders, assigning them to a specific provider, recording the requested amount of units and order date, and finally updating the order date and time and setting the state to 'placed'. The requested units are calculated to attain a maximum threshold (that is, the maximum stock minus the current stock). When the items of the replacement order are received, its row is updated with the correspondent receiving date, total (payment), and order state to 'fulfilled'. There can be only one non-fulfilled (either draft or placed) replacement order per reference, and under no circumstances will more than one order be placed per day and reference. When a new reference is inserted, the available stock is of course zero (unless the data operator who is inserting the row states explicitly something different). Next time the automatic process is run, it will set the order for filling the stocks up to the maximum threshold (by default, 10 units higher than the minimum stock). When placing each specific order, if at that moment there is no provider for that reference, the order will remain in 'draft' status. On the contrary, if there is more than one supplier for that reference, one will be chosen for that order. Specifically, the provider offering the lowest cost for the reference will be picked (if there are several tied offers, the supplier who has fulfilled their orders the fastest over the last year will be chosen; if they are still tied, the one with the fewest orders in the last year; if then still tied, anyone).

Draft orders can be deleted or updated (for increasing 'quantity'). Already placed orders can't be deleted, and shouldn't be updated (except to set the delivery date and change its status). Fulfilled orders can't be deleted or updated. The suppliers (providers) have a unique name, CIF (tax id), salesperson's full-name (cannot be repeated) and email (also univocal), phone number (unique) and commercial address (and, yes, it is unique as well). If the provider is removed from

the base, so will be all their supply lines (offers), but not the already fulfilled orders to them (which will be kept without a value for provider).

Besides, it is also needed to store the purchases placed by customers (our real business). An order is placed by a customer on a given date and set to a delivery address. They can set multiple purchases in the same day and to the same delivery address, but it will be processed as a single delivery (a delivery includes all orders of that day, and all items will be sent together). It is also important to record the billing data: type of payment (cash on delivery, bank transfer, credit card), payment date (charges to credit cards are always placed on the order's date), and credit card data (in case). The delivery data is also essential: full address, and delivery date. Finally, for each order there can be several items (one per purchased reference; in case that the customer buys more units of a certain product that had already purchased on the same day to the same shipping address, the new quantities will be added, keeping a single item line for that reference). Apart from the reference, it is also required the quantity, unit retail price (in the order's date) and total price. If there is not enough stock of a given reference (when placing the item of an order), the quantity will be set to the maximum available, and a message will be generated to inform the user of such event.

Our business distinguishes two types of clients: registered and non-registered clients. Registered clients have login data (unique username and password), while non-registered clients can only be identified by their contact data (either the email or the phone, since both attributes are unique, and all purchasers will always provide at least one of them). Regarding registered clients, apart from login data, we will keep their registration date and time; personal data (name and surname/s); contact information (phone and/or email); contact preference (email, phonecall, sms, whatsapp, facebook, wechat, qqmobile, snapchat, or telegram); their addresses (at least one address, and at most one address per client and town); their credit cards data; loyalty discounts; and, of course, all their purchases. Finally, non-registered users' data or any data related thereto won't be kept, except that concerning legitimate interest: data related to the purchases, including the references and quantities, the buyer's name and surname, shipping and billing addresses, payment data, credit card (in case), and relevant dates. If a registered user wants to be deregistered, all their purchases will be converted (to anonymous buyer's purchases) and then all data will be automatically deleted.

Data related to credit cards includes its cardholder, finance company, card number, and expiration date (only month and year); credit card numbers cannot be repeated, not even with different companies. In another vein, any address has the thoroughfare's type (street, square, circus, road...) and name, and may have gateway number, block number, stairs id, floor, and door. In addition, they always have ZIP code, town/city and country. Loyalty discounts are generally calculated monthly and recorded as a 'voucher' including the percentage to be discounted (on each future purchase) and the date (the voucher will only be valid for the following 30 days). That calculation follows a certain procedure (1% for every €10 purchased during the last 30 days, up to 10 times; plus the immediately previous discount divided by ten), and only current voucher is kept for each client. Regarding the contact preference, it is set by default to *sms* (short message) if there is a phone number for that client, and email otherwise.

Last (but not least), users can post their opinions, ratings and comments on products and references (if the comment is set on a product, it is assumed to be valid for all formats; otherwise, the format is specified). Comments have a score (a number from 1 to 5), a text, and can accumulate likes (supposed to be less than 1 billion, default zero). Comments can be tagged with an endorsement if they come from a registered user who has consumed that product/reference previously. If a user unsubscribes (is removed from the database) their comments remain, but anonymized (without link to any registered user). Unregistered customers cannot register any comments or ratings.

2. – STARTING POINT

The current database is extremely poor, with only three disjointed tables (and hardly any restrictions): a table registering products, and their suppliers (*catalogue*); another one regarding clients and orders (*trolley*); and a third one registering the user's comments on products (*posts*). The description of these tables is as follows:

```
SQL> desc dbf10.catalogue
```

Name	Type
BARCODE	CHAR(15)
PRODUCT	CHAR(50)
FORMAT	CHAR(20)
COFFEA	CHAR(20)
VARIETAL	CHAR(30)
ORIGIN	CHAR(15)
ROASTING	CHAR(10)
DECAF	CHAR(12)
PACKAGING	CHAR(15)
RETAIL_PRICE	CHAR(14)
COST_PRICE	CHAR(12)
CUR_STOCK	CHAR(5)
MIN_STOCK	CHAR(5)
MAX_STOCK	CHAR(5)
SUPPLIER	CHAR(35)
PROV_TAXID	CHAR(10)
PROV_BANKACC	CHAR(30)
PROV_ADDRESS	CHAR(120)
PROV_COUNTRY	CHAR(45)
PROV_PERSON	CHAR(90)
PROV_EMAIL	CHAR(60)
PROV_MOBILE	CHAR(9)

```
SQL> desc dbf10.trolley
```

Name	Type
ORDERDATE	CHAR(14)
ORDERTIME	CHAR(14)
PRODUCT	CHAR(50)
BARCODE	CHAR(15)
PRODTYPE	CHAR(20)
COFFEA	CHAR(20)
VARIETAL	CHAR(30)
ORIGIN	CHAR(15)

ENGINEERING PROGRAM FOR INTERNATIONAL STUDENTS

Academic course: 2024/2025 – 1st term

Subject: *Database Fundamentals*

Statement of 1st Assignment: Relational DB Design and Implementation

ROASTING	CHAR(10)
DCAFFPROCESS	CHAR(12)
PACKAGING	CHAR(15)
BASE_PRICE	CHAR(10)
DISCOUNT	CHAR(3)
QUANTITY	CHAR(2)
USERNAME	CHAR(30)
REG_DATE	CHAR(14)
REG_TIME	CHAR(14)
USER_PASSW	CHAR(15)
CLIENT_NAME	CHAR(35)
CLIENT_SURN1	CHAR(30)
CLIENT_SURN2	CHAR(30)
CLIENT_EMAIL	CHAR(60)
CLIENT_MOBILE	CHAR(9)
PAYMENT_TYPE	CHAR(15)
PAYMENT_DATE	CHAR(14)
PAYMENT_TIME	CHAR(14)
CARD_COMPANY	CHAR(15)
CARD_NUMBER	CHAR(20)
CARD_HOLDER	CHAR(30)
CARD_EXPIRATN	CHAR(7)
BILL_WAYTYPE	CHAR(10)
BILL_WAYNAME	CHAR(30)
BILL_GATE	CHAR(3)
BILL_BLOCK	CHAR(1)
BILL_STAIRW	CHAR(2)
BILL_FLOOR	CHAR(7)
BILL_DOOR	CHAR(1)
BILL_ZIP	CHAR(5)
BILL_TOWN	CHAR(45)
BILL_COUNTRY	CHAR(45)
DLIV_DATE	CHAR(14)
DLIV_TIME	CHAR(14)
DLIV_WAYTYPE	CHAR(10)
DLIV_WAYNAME	CHAR(30)
DLIV_GATE	CHAR(3)
DLIV_BLOCK	CHAR(1)
DLIV_STAIRW	CHAR(2)
DLIV_FLOOR	CHAR(7)
DLIV_DOOR	CHAR(2)
DLIV_ZIP	CHAR(5)
DLIV_TOWN	CHAR(45)
DLIV_COUNTRY	CHAR(45)

```
SQL> desc dbf10.posts
```

Name	Type
-----	-----
USERNAME	CHAR(30)
PRODUCT	CHAR(50)
BARCODE	CHAR(15)
POST_DATE	CHAR(14)
POST_TIME	CHAR(14)
TITLE	CHAR(50)
TEXT	CHAR(2000)
SCORE	CHAR(1)
LIKES	CHAR(9)
ENDORSED	CHAR(50)

3. – SUPPORTING MATERIALS

Apart from classes and tutoring sessions, students can count on the following resources:

- Documents: assignment statement (this doc); class slides; template for writing the assignment report (.docx format)
- Audiovisual resources: video classes to acquire specific knowledge about the use of the tools that will be used in the laboratories (console management and pl/sql syntax) in the “inverted class” style.
- Sw Resources: user account on RDBMS Oracle (accessible from all computer rooms in the University, and from [Aula Virtual](#)), with enough privileges for all required operations and reading privileges on the obsolete DB’s tables.

4. – TO DO

- a) Do an adequate Relational Design meeting the requirements (as possible). Represent it by means of a relational graph (use notation provided in class) and accompany with relevant comments about the non-observed explicit semantics (each assumption not reflected will have an identifier or a number to properly reference it in other sections), and the implicit semantic assumptions that have been included in the design.
- b) Implement that design in PL/SQL (on DBMS Oracle®). The simple restrictions deemed appropriate will be included, indicating the explicit semantics they incorporate. Likewise, the new implicit/explicit-excluded semantic assumptions that appear in this phase will be documented.
- c) Do a massive data upload (import) from the old DB to the newly created one. Notice there could appear errors during this process, due to the lack of some attributes and/or constraints (poor semantic coverage) in the former database. Coming from insufficiently defined tables, data may present inconsistencies, lack of integrity, and data errors in general. You must detect, analyze, typify, and describe them in the report. If possible, propose and implement some solution.
- d) Document all the work carried out by means of the pertinent *Labwork Report*, for which composition a template is provided. Apart from including the relational graph and semantic comments, make sure that all design decisions are conveniently justified and thus reflected in the report.
- e) Submit through Aula Global a compressed .zip file containing three files: two scripts (creation.sql, upload.sql) and the “lab work report” saved as .pdf (portable document format), and named ***nia1_nia2_nia3_LW1.pdf*** (where ‘nia’ is the students’ identifier).