

# GBD03 Task

## Wording.

### EXERCISE 1

Writes SQL instructions necessary to create the database from the following relational model that corresponds to the Mouro Restaurant:

EMPLOYEES (ID, Name, LastName, Address, City, Phone, CP, EntryDate, Category, Salary)  
 COOKS (ID, Title, Specialty)  
 ADMINISTRATION (ID, Position)  
 WAITERS (ID, Turn, years, *incharge\_ID*)  
 DININGROOM (CodeD, Name, Capacity, NTables, Location)  
 TABLES (CodeT, *CodeD*, Nseats, *waiter\_ID*)  
 Make (*CodeT*, *CodeD*, *Nreservation*, NseatsR)  
 RESERVATIONS (*Nreservation*, Dtreservation, Name, Dtreserved, lunch / dinner, Npeople, Various)  
 BILLS (*NBill*, DateB, *CodeT*, *CodeD*)  
 Include (*NBill*, *CodeDs*, Units)  
 DISHES (*CodeDs*, Name, Description, Type, Price)  
 Contain (*CodeDs*, *CodePr*, Amount)  
 PRODUCTS (*CodePr*, Description, Stock, BaseUnit, PriceU, Category, *CodeSp*)  
 SUPPLIERS (*CodeSp*, Address, Zip, Phone, Fax, Contact)

**Note:** Primary keys are underlined and foreign keys are denoted in italics.

### EXERCISE 2

About the database Mouro Restaurant made the following changes:

#### Table EMPLOYEES

- 2.1. Add an index to facilitate frequent searches by last name without duplicates.
- 2.2. The restaurant has opened on July 15, 2010. Check that the entry date of the employees is not earlier.

#### Table WAITERS

- 2.3. The turn can only take 3 values: morning, noon and night. Add this restriction considering that a waiter can have more than one turn (do not use CHECK).

#### Table SUPPLIERS

- 2.4. Add the columns Name and Surname between the code and the address.

#### Table TABLES

- 2.5. The default number of seats at the tables is 4.

#### Table PRODUCTS

- 2.6. Add an index by category. Displays all indexes of the table.
- 2.7. Add a constraint on the table, so that the Stock is 4-digit integer, unsigned and not admit null values.
- 2.8. Delete the newly created index.

#### Table DISHES

- 2.9. Deletes the table SUPPLIERS. What happen? Deletes foreign keys previously.

#### Mouro Restaurant DATABASE

- 2.10. Deletes the database.

**Scoring criteria. Total 10 points.**

Total score: 10 points distributed as follows:

#### Exercise 1: 5 points:

- Successful SQL writing basic tables.
- Successful SQL writing of the relationships between tables.
- Correct election of data types and sizes.

- Include the necessary restrictions.

**Exercise 2: 5 points:**

- 0.5 points for each section.

**Required resources to perform the task.**

To perform this task you need to write SQL statements. You can use:

- The notebook application.
- A SQL editor installed on a DBMS. We recommend using PhpMyAdmin or MySQL Workbench that are used in the examples of the unit, but you can choose any other.

**Tips and recommendations.**

- The advantage of using the SQL editor with a graphic tool is that you can check if you have any errors in your code before running the script.
- When the SQL editor will display an error, carefully read the message.
- Use different lines for the definition of each attribute so you'll know better where the error is.
- Take care with the sequence of the creation of tables when there are foreign keys references.
- Verify that the data types and sizes are appropriate to the attribute defined.
- Remember that the data types and sizes must match the keys that relate.
- Do not forget to choose the right table type to work with foreign keys.
- Save the script to reload again if you damage the tables while you are trying SQL sentences.

**Indicaciones de entrega.**

Una vez realizada la tarea elaborarás un único documento dónde figuren las respuestas correspondientes. El envío se realizará a través de la plataforma de la forma establecida para ello, y el archivo se nombrará siguiendo las siguientes pautas:

**apellido1\_apellido2\_nombre\_GBD03\_Tarea**

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna **Begoña Sánchez Mañas para la tercera unidad del MP de GBD**, debería nombrar esta tarea como...

**sanchez\_manas\_begona\_GBD03\_Tarea****Delivery Indications.**

Once the task are finished, put it in a document with the answers. Delivery will be made through the platform as established for it, and the file will be named using the following guidelines:

**surname1\_surname2\_name\_GBD03\_Task**

Make sure the name does not contain special characters. For example the student **Begoña Sánchez Mañas for the third unit of the MP of GBD**, should appoint this task as ...

**sanchez\_manas\_begona\_GBD03\_Task**