

COMP9120 Database Management Systems**Assignment 2: Database Application Development****Group Assignment (15%)****Introduction**

The objectives of this assignment are to gain practical experience in interacting with a relational database management system using an Application Programming Interface (API) (Python DB-API). This assignment additionally provides an opportunity to use more advanced features of a database such as functions.

This is a group assignment for teams of 3 members. It is assumed that you will continue in your Assignment 1 group. You should inform the unit coordinator as soon as possible if you wish to change groups.

Please also keep an eye on your email and any announcements that may be made on Ed.

Submission Details

The final submission of your database application is due at 11:59pm on Sunday 12th May (Week 11). You should submit the *items for submission* (detailed below) via Canvas.

Items for submission

Please submit your solution to Assignment 2 in the 'Assignment' section of the unit's Canvas site by the deadline, including EXACTLY THREE files:

- An assignment coversheet as a PDF document (.pdf file suffix) which is available for download from [this link](#) on Canvas.
- A SQL file (**FFKSchema.sql**) containing the SQL statements necessary to generate the database schema and sample data. This must contain the original schema and insert SQL statements, and any stored procedures (functions) you may have added.
- A Python file (**database.py**) containing the Python code you have written to access the database.

Task 1: The Fine Food Kitchen (FFK) System

In this assignment, you will be working with the FFK system for staff use that you will be developing. The system requires work in numerous areas, including interaction with the database. Your main task in this assignment is to handle requests for *reads* and *writes* to the database using your user interface (UI). We first describe the main features that the FFK system should include from a UI perspective, and then discuss where the majority of your database code needs to be implemented.

Logging In

The first form a user is presented with, when starting the FFK system is the Login, as shown in Figure 1. This feature requires that a staff enters their username and password to be validated prior to being successfully logged in to the system. Security features such as password encryption/hashing is out of scope for this assignment. Once logged in, the user is taken to the menu items list page to view their associated menu items.

Log in

Log in

Figure 1 – Login form

Viewing Menu Items List

Once a user is logged in, they are shown a list of all their associated menu items, as shown below in Figure 2. This list must be ordered such that the most recently reviewed menu items appear at the bottom. The list is also sorted by description in ascending order, and price in descending order. Each menu item has a name, description, category, coffee option (coffee type, milk kind), price, review date, and reviewer. A menu item is associated with a staff if the staff is assigned as the person in charge of reviewing the menu item.

Fine Food Kitchen

Menu ItemsAdd Menu ItemLogout

Menu Items

Find

ID	Name	Description	Category	Option	Price	Review Date	Reviewer
1	French Toast	A sliced bread soaked in beaten eggs, milk, and cream, then pan-fried with butter	Breakfast		9.90	10-01-2024	John Doe
8	Seafood Cleopatra	Salmon topped with prawns in a creamy seafood sauce. Served with salad and chips	Dinner		25.90	20-02-2024	John Doe
7	Coffee Drink		Breakfast Lunch	Cappuccino - Soy	3.50	28-02-2024	John Doe
3	Poke Bowl	Cubes of marinated fish tossed over rice and topped with vegetables and Asian-inspired sauces	Breakfast Lunch Dinner		15.90	28-02-2024	John Doe

Figure 2 – Viewing Menu Items List

Finding Menu Items

A user can search through all menu items by entering a word or phrase (a ‘keyword’) in the field next to the Find button, as shown in Figure 3, and then clicking on Find. When such a keyword is specified, then the retrieved menu items and shown on the list are those that include this word or phrase in the name, description, category, coffee option (coffee type or milk kind), or reviewer’s full name. The search is case insensitive. For example, given the search keyword ‘fee’, Find will return all menu items that include the keyword ‘fee’ in the name, description, category, coffee option (coffee type or milk kind), or reviewer’s full name. Searching with a blank/empty keyword field will show all of the logged in user’s associated menu items. Any search results returned must be ordered such that menu items without a reviewer would appear at the top, and then by review date in descending order. The search results must exclude those menu items

whose review dates are older than 10 years (from today’s date).

Fine Food Kitchen

Menu Items

Add Menu Item

Logout

Menu Items

fee

Find

ID	Name	Description	Category	Option	Price	Review Date	Reviewer
9	Iced Coffee	A glass of cold espresso, milk, ice cubes, and a scoop of ice cream	Breakfast Lunch	Latte - Whole	7.60		
6	Black Coffee		Breakfast Lunch Dinner	LongBlack	4.30		
5	White Coffee	A full-flavored concentrated form of coffee served in small, strong shots with whole milk	Breakfast Lunch	Espresso - Whole	3.50	22-03-2024	Robert Tatum
7	Coffee Drink		Breakfast	Cappuccino - Soy	3.50	28-02-2024	John Doe

Figure 3 – Finding Menu Items

Adding a Menu Item

Users may also add a new menu item by clicking on the *Add Menu Item* tab in the title bar. They may enter menu item details such as name, description, categories, coffee type (for coffee item), milk kind (for coffee item), and price, in the popup dialog. Once the popup dialog appears, click on Add Menu Item button, as shown in Figure 4. A new menu item will not be assigned a review date nor reviewer. A coffee type must be provided when a milk kind is specified.

Fine Food Kitchen

Menu Items

Add Menu Item

Logout

New Menu Item

Name:

Coffee Drink

Description:

A coffee drink comprising espresso, steamed milk, and foam

Category 1:

breakfast

Category 2:

LUNCH

Category 3:

Coffee Type:

Cappuccino

Milk Kind:

OAT

Price:

5.70

Add Menu Item

Figure 4 – Adding a Menu Item

Updating a Menu Item

Users can also update a menu item by modifying data in the menu item details screen as shown in Figure 5, and clicking on Update Menu Item button. You can access this update screen by clicking on a menu item from the list of menu items in the Menu Items tab.

Fine Food Kitchen Menu Items Add Menu Item Logout

Update Menu Item

ID:

Name:

Description:

Category 1:

Category 2:

Category 3:

Coffee Type:

Milk Kind:

Price:

Review Date:

Reviewer:

Update Menu Item

Figure 5 – Updating a Menu Item

Database Interaction Code

The files that are needed for the Python version of assignment are as follows:

1. **FFKSchema.sql**: a file which contains SQL statements you need to run to create and initialise the FFK system database, before starting the application
https://canvas.sydney.edu.au/files/36452494/download?download_frd=1
2. **Assignment2_PythonSkeleton.zip**: a zip file encapsulating the Python project for the FFK system
https://canvas.sydney.edu.au/files/36393167/download?download_frd=1

To inspect the FFK system code, you need to unzip the ZIP archive first, which will create a folder that includes the name **Assignment2_PythonSkeleton**. If you experience any difficulties installing and exploring the project, ask your tutor or lecturer for assistance.

The skeleton code uses a number of Python modules to implement a simple browser-based GUI for the FFK system. The main modules are the Flask framework for the GUI and the psycopg2 module for the PostgreSQL database access. Similar to tutorial 6, **you will need to install the Psycopg2 module and the Flask module**. The skeleton code follows the structure described below:

- The main program starts in the **main.py** file. You need to use the correct username/password details as specified in tutorial 6, and then implement the missing database access functions – including any necessary SQL code statements required – in the data layer **database.py**.
- The presentation layer is done via a simple HTML interface that can be accessed from a web browser. The corresponding page templates are located in the **templates/** subdirectory, their CSS style file

is `static/css`.

- The transition between the different GUI pages and the initialisation of the Flask framework is done in the `routes.py` file. It currently just invokes the pages, but there is no further business logic implemented yet.

You can run the code by running “`python main.py`”. This starts a local web server and prints out some debug messages in the terminal; the GUI can then be accessed with any web browser on the same computer via the local URL <http://127.0.0.1:5000/> (If that doesn't work you can also try <http://0.0.0.0:5000/>). Please note that, to terminate the application, you will need to stop the local web server which is running in the background.

Task 2: Functions Implementation

Core Functionality

In this assignment, you are provided with a Python skeleton project that must serve as the starting point for your assignment. Your task is to provide a complete implementation for the file `database.py`, as well as make any modifications necessary to the database schema (i.e., `FFKSchema.sql`). Specifically, you need to modify and complete these five functions:

1. `checkStaffLogin` (for login)
2. `findMenuItemsByStaff` (for viewing menu items list)
3. `findMenuItemsByCriteria` (for finding menu items)
4. `addMenuItem` (for adding menu item)
5. `updateMenuItem` (for updating menu item)

Note that, for each function, the corresponding action and outcome should be implemented by issuing SQL queries to the database management system. If you directly output the result set, pre-process, manipulate and/or make changes to the input or output datasets using Python code or additional modules (libraries) i.e. without issuing SQL queries, you are considered as cheating, and you will get penalised heavily and most likely get zero point for the assignment. No additional Python modules or libraries should be imported.

Marking

This assignment is worth 15% of your final grade for the unit of study. Your group's submission will be marked in line with the rubric that follows.

Group member participation

If members of your group do not contribute sufficiently, you should alert the unit coordinator as soon as possible. The course instructor has the discretion to scale the group's mark for each member as follows:

Percentage of contribution	Proportion of final grade received
< 5% contribution	0%
5 - 10% contribution	20%
11 - 15% contribution	40%
16 - 20% contribution	50%

21 - 24% contribution	60%
25 - 28% contribution	80%
29 - 30% contribution	90%
> 30% contribution	100%

Note: The above table assumes that each group will have 3 members, so, on average, around 33% contribution is expected from each member of the group. In special case, if the group has less than 3 members then the contribution percentage will be adjusted accordingly. You must justify your contribution percentage by providing detailed explanation of your contribution on the coversheet mentioned before. You should also maintain a diary of your group meetings and discussions on Canvas. Furthermore, we will run random face-to-face interviews to understand your contribution, if needed.

Marking Rubric

Your submissions will be marked according to the following rubric, with a maximum possible score of 15 points.

	Part Marks (0 – 1.5 pts)	Full Marks (2 – 2.5 pts)
Login	Can correctly login the user 'JOHNDOE' and validate his username and password.	All valid users can be logged in successfully, and unsuccessful user logins should be rejected.
View Menu Items List	Correctly list all menu items associated with user 'johndoe' in the correct order (see Figure 2).	Correctly list all menu items associated with a user, in the correct order, for all possible username input from Figure 1.
Find Menu Item	Correctly list menu items for keyword "fee" (see Figure 3).	Correctly list menu items for all possible keywords.
Add Menu Item	Can correctly add a menu item to the database.	Can correctly add all valid menu items to the database. Menu items entered with invalid details should be rejected.
Update Menu Item	Can correctly update the status of a menu item.	Can correctly update details of all menu items, ensuring the updated details for a menu item are valid.
Stored Procedure (Function)	A couple of stored procedures (functions) are correctly created in the submitted SQL file.	A couple of stored procedures (functions) are correctly created in the submitted SQL file, and correctly called in two of the five specified functions.

	No Marks (0 pt)	Full Marks (1 pt)
Record Keeping of Group Discussions	One or more issues reported or found with group member participation and maintaining records of group meetings and discussions on Canvas.	No issue reported or found with group member participation. All group members participate and regularly maintain a diary of group meetings and discussions on Canvas.