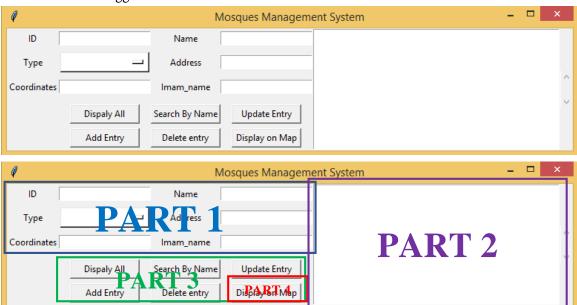


المملكة العربية السعودية وزارة التعليم كلية الحاسب قسم علوم الحاسب

CS492 443-1444/2023 Project 3 Deadline: Week 11

## Develop a simple Mosques Management System considering the following:

A. The system should be developed with a Graphical User Interface using TKinter This is a suggested structure of the interface



The interface contains the following elements:

Part 1: input fields to allow the users to enter mosques data as follows:

Field Name	Data Type	TK Widget Type	
ID	Number	Entry	
Name	Text	Entry	
Туре	Text	OptionMenue	
Address	Text	Entry	
Coordinates	Text	Entry	
Imam_Nmae	Text	Entry	

- Part 2: A ListBox to display the data records
- Part 3: selection buttons to allow the user selecting the required operation

Button Title	Operation	
Display_All	to obtain the records of all the mosques from the database and	
	display them on the ListBox	
Search	to obtain one record from the database using the mosque name as	
	a search key, and then display the search output on the ListBox	
Add_Entry	to insert a record in the database for a new mosque by sending	
	user entries at the input fields (Part 1) to the database	
Delete_Entry	to delete one record from the database using the mosque ID	

■ Part 4: These are for the extra work that can be added (see below)

- B. It is required to build a class for the mosques and then create a mosque object for each entry
- C. The system should be connected to a database to contain mosques data in the Mosq table (use sqlite3 library)

Build a class for the database to include the following methods:

Method	Description	Argument	Return
init()	to connect to the database and then		
	create the Mosques table once a new	None	None
	object is created		
Display()	to fetch and return all the records	None	All the
	from the table (use SELECT query)	None	records
Search(name)	to fetch and return one record that	nama	one record
	match the given key word (name)	name	
Insert(ID, name,		ID, name,	
type, address,	to add a record to the table with the	type, address,	None
coordinates,	given data	coordinates,	None
Imam_name)		Imam_name	
Delete(ID)	to delete one record that matches	ID	None
	the given input (ID)	ID	
del()	On destroying the object, you need	None	None
	to close the connection to DB	TNOILE	

Use the class to create a new database object and then use its methods for supporting all the operations of the system (displaying all the mosques records, searching for a record of a specific mosque, adding a new record for a mosque, deleting a record of a mosque)

- D. There are some extra features that you can incorporate into your system for additional marks:
  - 1. Adding the *Update* operation:

To enable the user to update the Imam\_name field in the database. In this case, the user firstly searches for a mosque record by name and then uses the additional 'update' button (part 4) to apply the modification after modifying the field.

2. Adding the *Display on Map* operation:

To enable displaying the location of the mosque on the map. In this case, the user first searches for a mosque record by name and then use the additional 'Display on Map' button (part 4) to display the location on the map using the *coordinates* information. *Hint*: use "folium" and "webbrowser" libraries.

3. Enhancing the search operation:

In case the user enters a misspelled name, the system can provide the user with the very close matching names in the database. Then the user can select the correct name to search for. *Hint*: use "difflib" library.