```
import tkinter as tk
from tkinter import ttk
from tkinter import messagebox
import sqlite3 as sql

def add_task():
    task_string = task_field.get()
    if len(task_string) == 0:
        messagebox.showinfo('Error', 'Field is Empty.')
    else:
        tasks.append(task_string)
        the_cursor.execute('insert into tasks values (?)', (task_string
,))
        list_update()
        task_field.delete(0, 'end')

def list_update():
    clear_list()
    for task in tasks:
        task_listbox.insert('end', task)

def delete_task():
    try:
        the_value = task_listbox.get(task_listbox.curselection())
        if the_value in tasks:
            tasks.remove(the_value)
            list_update()
            the_cursor.execute('delete from tasks where title = ?',
(the_value,))
    except:
        messagebox.showinfo('Error', 'No Task Selected. Cannot Delete.')

def delete_all_tasks():
    message_box = messagebox.askyesno('Delete All', 'Are you sure?')
    if message_box == True:
        while(len(tasks) != 0):
            tasks.pop()
        the_cursor.execute('delete from tasks')
        list_update()

def clear_list():
    task_listbox.delete(0, 'end')

def close():
    print(tasks)
    guiWindow.destroy()

def retrieve_database():
    while(len(tasks) != 0):
        tasks.pop()
    for row in the_cursor.execute('select title from tasks'):
        tasks.append(row[0])

if __name__ == "__main__":
```

```
guiWindow = tk.Tk()
guiWindow.title("To-Do List Manager - ARSHAD")
guiWindow.geometry("500x450+750+250")
guiWindow.resizable(0, 0)
guiWindow.configure(bg = "#FAEBD7")

the_connection = sql.connect('listOfTasks.db')
the_cursor = the_connection.cursor()
the_cursor.execute('create table if not exists tasks (title text)')


tasks = []

header_frame = tk.Frame(guiWindow, bg = "dark orange")
functions_frame = tk.Frame(guiWindow, bg = "dark orange")
listbox_frame = tk.Frame(guiWindow, bg = "dark orange")

header_frame.pack(fill = "both")
functions_frame.pack(side = "left", expand = True, fill = "both")
listbox_frame.pack(side = "right", expand = True, fill = "both")

header_label = ttk.Label(
    header_frame,
    text = "To-Do List",
    font = ("Alice", "30", "bold"),
    background = "dark orange",
    foreground = "#FFFFFF"
)
header_label.pack(padx = 10, pady = 10)

task_label = ttk.Label(
    functions_frame,
    text = "Enter the Task:",
    font = ("Alice", "11", "bold"),
    background = "dark orange",
    foreground = "#FFFFFF"
)
task_label.place(x = 30, y = 40)

task_field = ttk.Entry(
    functions_frame,
    font = ("Consolas", "12"),
    width = 18,
    background = "dark orange",
    foreground = "#A52A2A"
)
task_field.place(x = 30, y = 80)

add_button = ttk.Button(
    functions_frame,
    text = "Add Task",
    width = 24,
    command = add_task
)
```

```
del_button = ttk.Button(
    functions_frame,
    text = "Delete Task",
    width = 24,
    command = delete_task
)
exit_button = ttk.Button(
    functions_frame,
    text = "Exit",
    width = 24,
    command = close
)
add_button.place(x = 30, y = 120)
del_button.place(x = 30, y = 160)
exit_button.place(x = 30, y = 200)

task_listbox = tk.Listbox(
    listbox_frame,
    width = 26,
    height = 13,
    selectmode = 'SINGLE',
    background = "#FFFFFF",
    foreground = "#000000",
    selectbackground = "#CD853F",
    selectforeground = "#FFFFFF"
)
task_listbox.place(x = 10, y = 20)

retrieve_database()
list_update()
guiWindow.mainloop()
the_connection.commit()
the_cursor.close()
```