SCHOOL OF INFORMATION SCIENCE

COLLEGE OF COMPUTING, INFORMATICS AND MEDIA

UNIVERSITI TEKNOLOGI MARA

MERBOK, KEDAH

DIPLOMA IN INFORMATICS LIBRARY (IM144)

PROGRAMMING FOR LIBRARIES (IML208)

"**INDIVIDUAL PROJECT: COMPUTER PROGRAM FOR MUSIC CLASS REGISTRATION**"

BY:

NURUL FITRIYAH BINTI MOHD SHOKRI (2022858108)

CLASS: KCDIM1443E

PREPARED FOR:

AIRUL SHAZWAN BIN NORSHAHIMI

SUBMISSION DATE:

4 JANUARY 2024

INDIVIDUAL ASSIGNMENT: COMPUTER PROGRAM FOR MUSIC CLASS REGISTRATION

BY:

NURUL FITRIYAH BINTI MOHD SHOKRI

(2022858108)

SCHOOL OF INFORMATION SCIENCE

COLLEGE OF COMPUTING, INFORMATICS AND MEDIA

UNIVERSITI TEKNOLOGI MARA

MERBOK, KEDAH

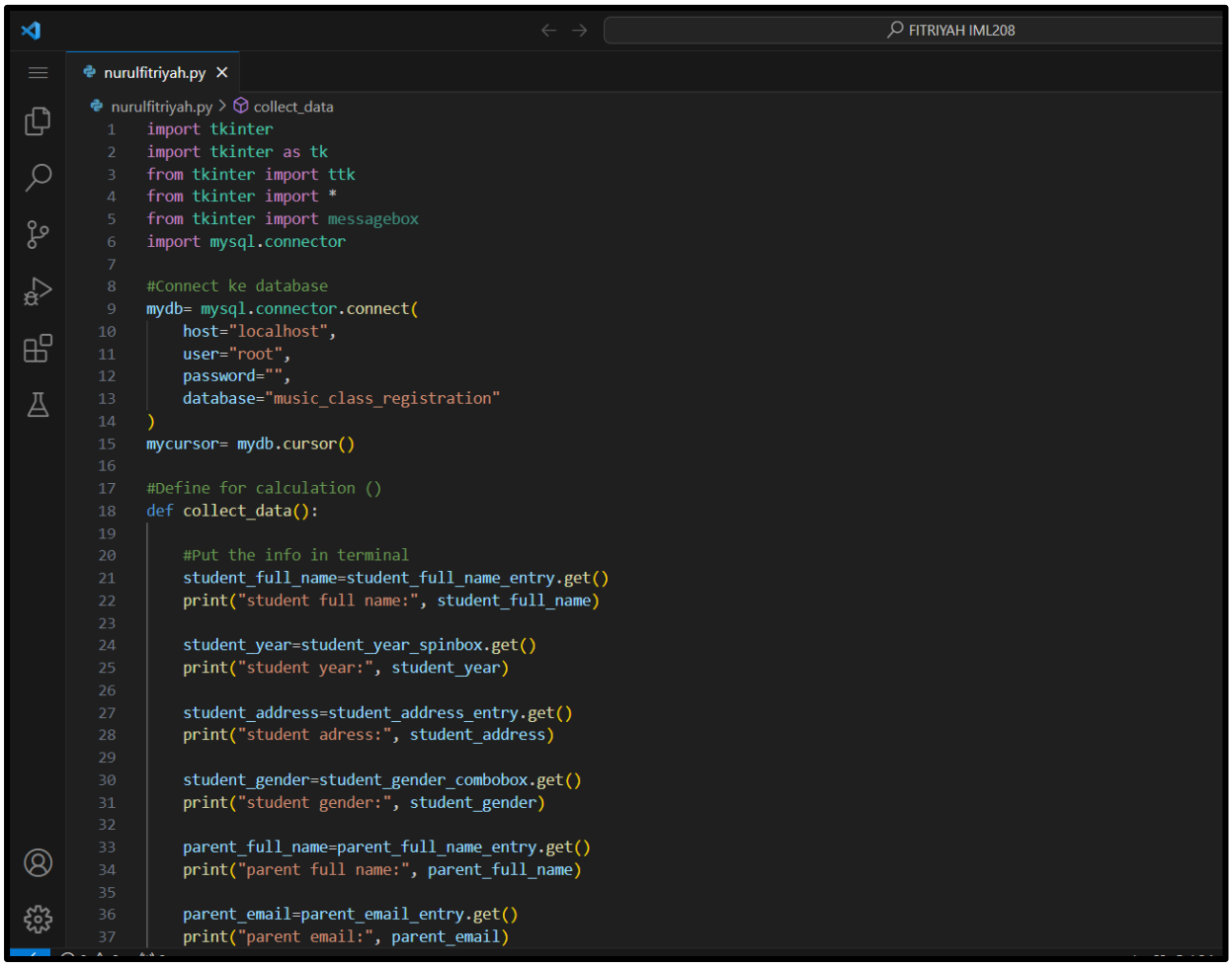
22 NOVEMBER 2023

1.0 INTRODUCTION

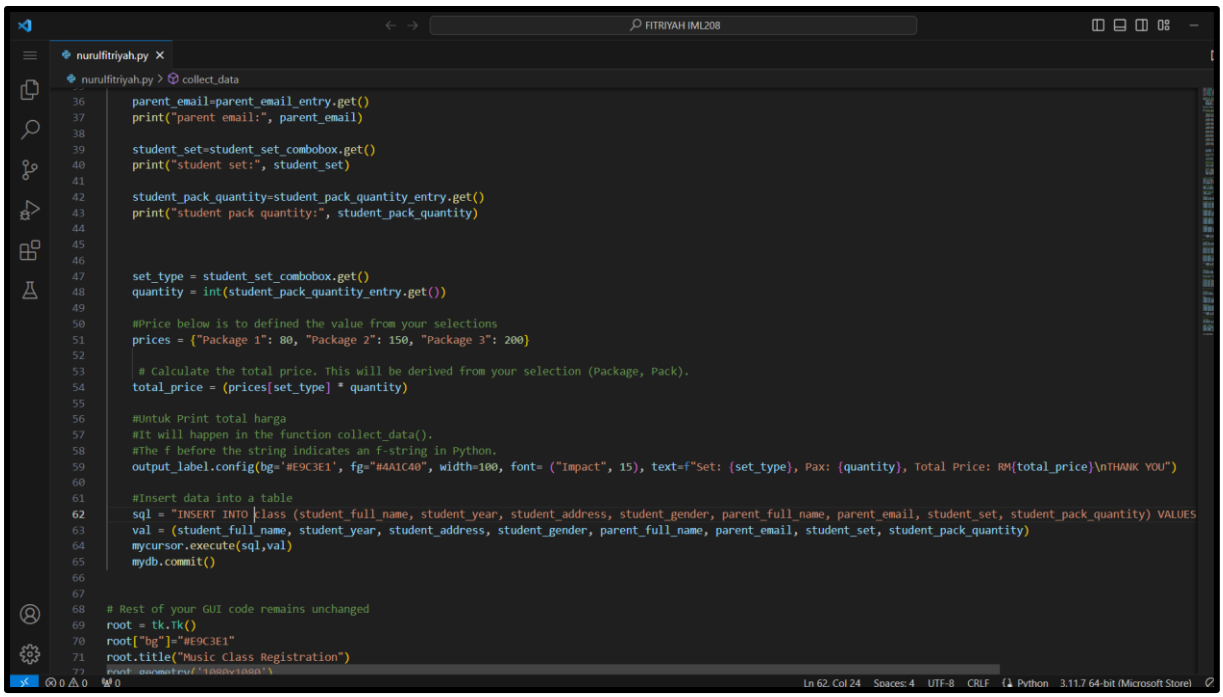
My brief computer program introduction consists of a Graphical User Interface (GUI) using Python Code and linking it to MySQL database. My GUI title is Music Class Registration which consists of information about the student's name, year, address, gender, guardian name, guardian email, package, quantity, and total price. I put calculations on my GUI Python Code to facilitate users to know the price that they selected. How this function works is when the user selects the package that they want, and if they want to change the quantity of i they can tick and the calculation will happen in Visual Studio code. The price of the package type will be multiplied by quantity and it will auto-generate in the price label. But if the user only chooses for package type without deciding the quantity, it will never show the total price and you can not submit the data.

Other than that, I also find the right color for my button, label, and input background to make it suitable, and make it well-organized, so the users can distinguish which part to fill in the information and which part to send data.

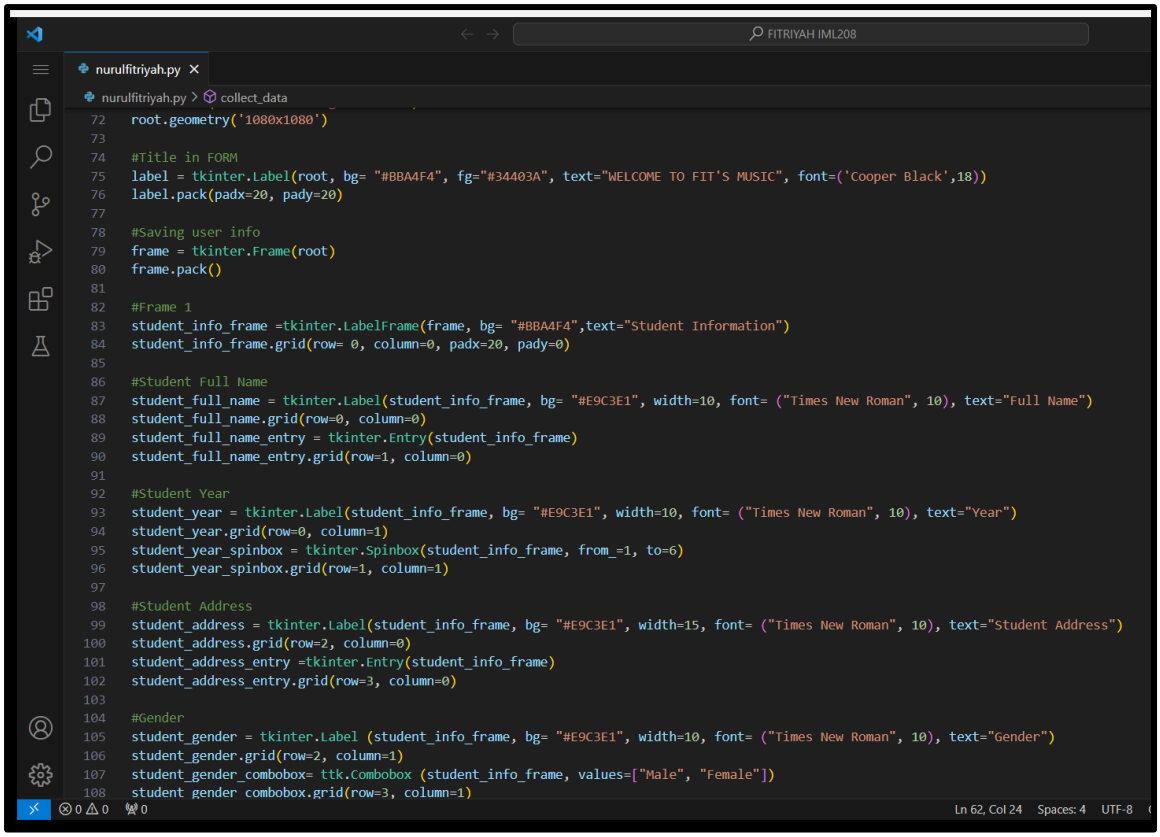
2.0 SOURCE CODE

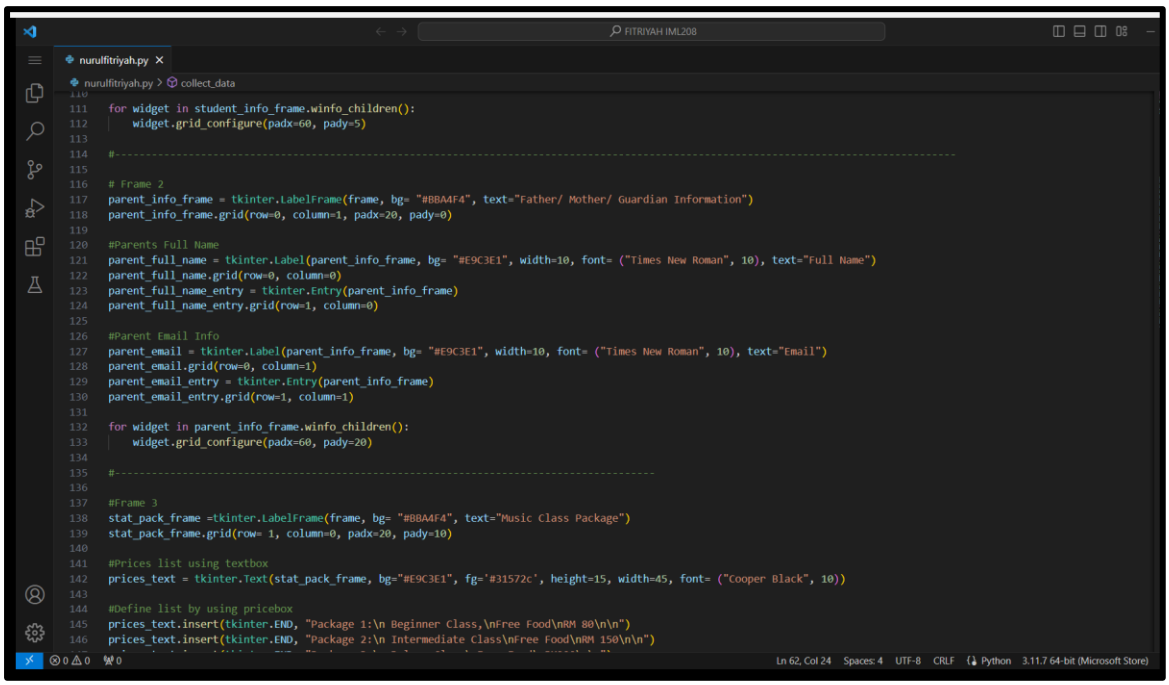


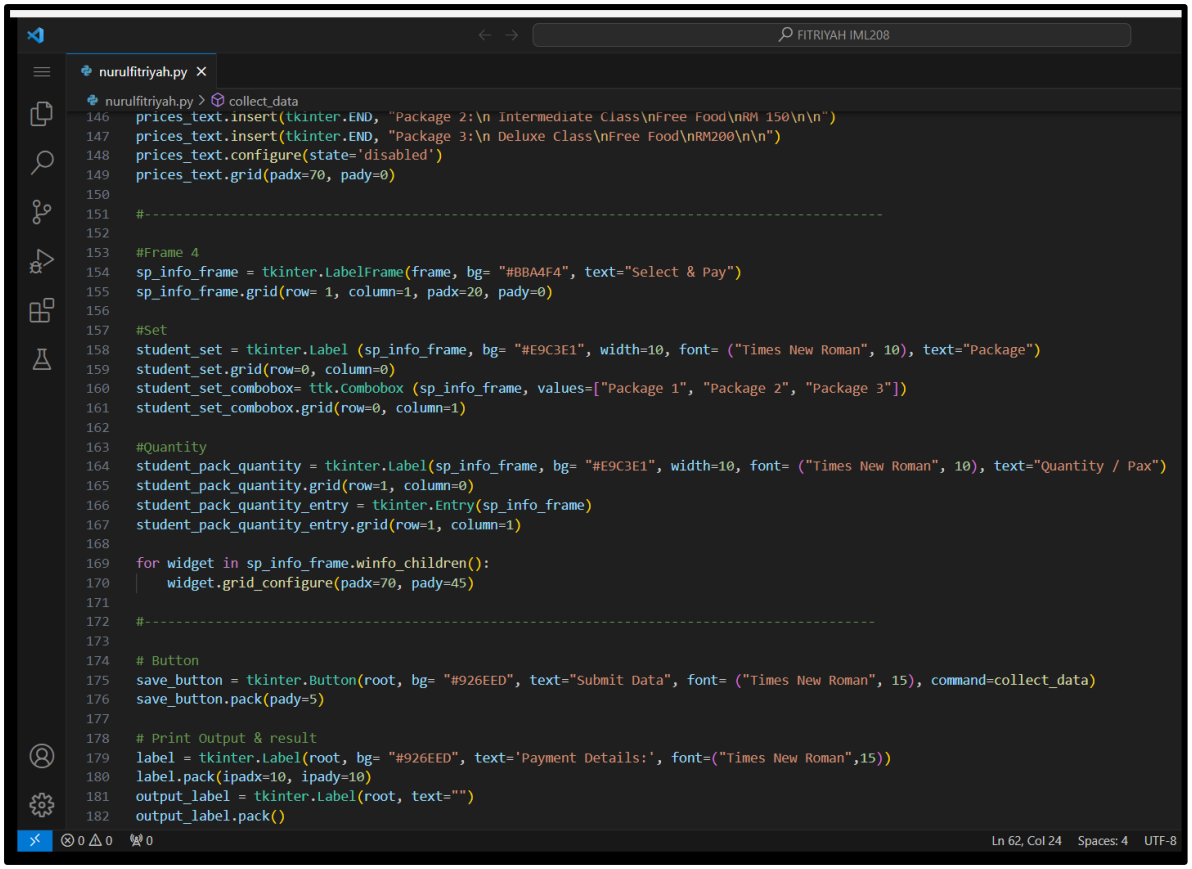
```python
import tkinter
import tkinter as tk
from tkinter import ttk
from tkinter import *
from tkinter import messagebox
import mysql.connector

#Connect ke database
mydb= mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="music_class_registration"
)
mycursor= mydb.cursor()

#Define for calculation ()
def collect_data():

    #Put the info in terminal
    student_full_name=student_full_name_entry.get()
    print("student full name:", student_full_name)

    student_year=student_year_spinbox.get()
    print("student year:", student_year)

    student_address=student_address_entry.get()
    print("student adress:", student_address)

    student_gender=student_gender_combobox.get()
    print("student gender:", student_gender)

    parent_full_name=parent_full_name_entry.get()
    print("parent full name:", parent_full_name)

    parent_email=parent_email_entry.get()
    print("parent email:", parent_email)
```



```python
    parent_email=parent_email_entry.get()
    print("parent email:", parent_email)

    student_set=student_set_combobox.get()
    print("student set:", student_set)

    student_pack_quantity=student_pack_quantity_entry.get()
    print("student pack quantity:", student_pack_quantity)



    set_type = student_set_combobox.get()
    quantity = int(student_pack_quantity_entry.get())

    #Price below is to defined the value from your selections
    prices = {"Package 1": 80, "Package 2": 150, "Package 3": 200}

    # Calculate the total price. This will be derived from your selection (Package, Pack).
    total_price = (prices[set_type] * quantity)

    #Untuk Print total harga
    #It will happen in the function collect_data().
    #The f before the string indicates an f-string in Python.
    output_label.config(bg='#E9C3E1', fg="#4A1C40", width=100, font= ("Impact", 15), text=f"Set: {set_type}, Pax: {quantity}, Total Price: RM{total_price}\nTHANK YOU")

    #Insert data into a table
    sql = "INSERT INTO class (student_full_name, student_year, student_address, student_gender, parent_full_name, parent_email, student_set, student_pack_quantity) VALUES
    val = (student_full_name, student_year, student_address, student_gender, parent_full_name, parent_email, student_set, student_pack_quantity)
    mycursor.execute(sql,val)
    mydb.commit()


# Rest of your GUI code remains unchanged
root = tk.Tk()
root["bg"]="#E9C3E1"
root.title("Music Class Registration")
```

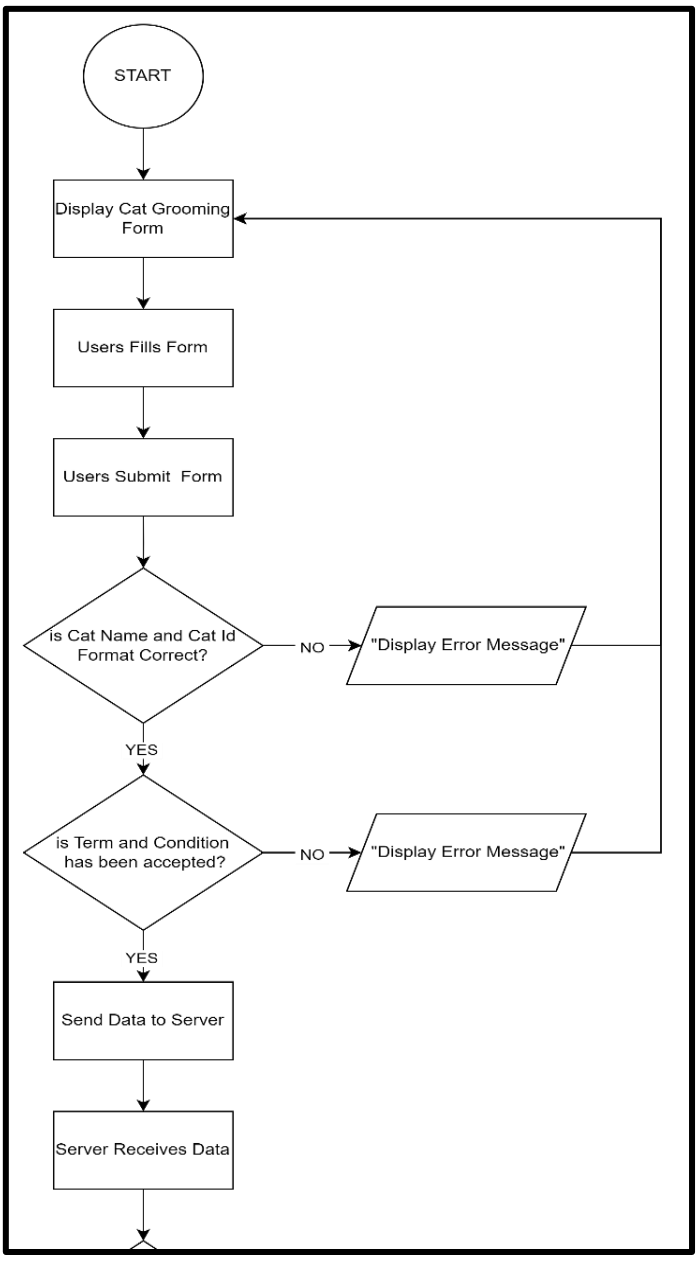

FITRIYAH IML208
nurulfitriyah.py
nurulfitriyah.py > collect_data
root.geometry('1080x1080')
#Title in FORM
label = tkinter.Label(root, bg= "#BBA4F4", fg="#34403A", text="WELCOME TO FIT'S MUSIC", font=('Cooper Black',18))
label.pack(padx=20, pady=20)
#Saving user info
frame = tkinter.Frame(root)
frame.pack()
#Frame 1
student_info_frame =tkinter.LabelFrame(frame, bg= "#BBA4F4",text="Student Information")
student_info_frame.grid(row= 0, column=0, padx=20, pady=0)
#Student Full Name
student_full_name = tkinter.Label(student_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Full Name")
student_full_name.grid(row=0, column=0)
student_full_name_entry = tkinter.Entry(student_info_frame)
student_full_name_entry.grid(row=1, column=0)
#Student Year
student_year = tkinter.Label(student_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Year")
student_year.grid(row=0, column=1)
student_year_spinbox = tkinter.Spinbox(student_info_frame, from_=1, to=6)
student_year_spinbox.grid(row=1, column=1)
#Student Address
student_address = tkinter.Label(student_info_frame, bg= "#E9C3E1", width=15, font= ("Times New Roman", 10), text="Student Address")
student_address.grid(row=2, column=0)
student_address_entry =tkinter.Entry(student_info_frame)
student_address_entry.grid(row=3, column=0)
#Gender
student_gender = tkinter.Label (student_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Gender")
student_gender.grid(row=2, column=1)
student_gender_combobox= ttk.Combobox (student_info_frame, values=["Male", "Female"])
student_gender_combobox.grid(row=3, column=1)
Ln 62, Col 24
Spaces: 4
UTF-8

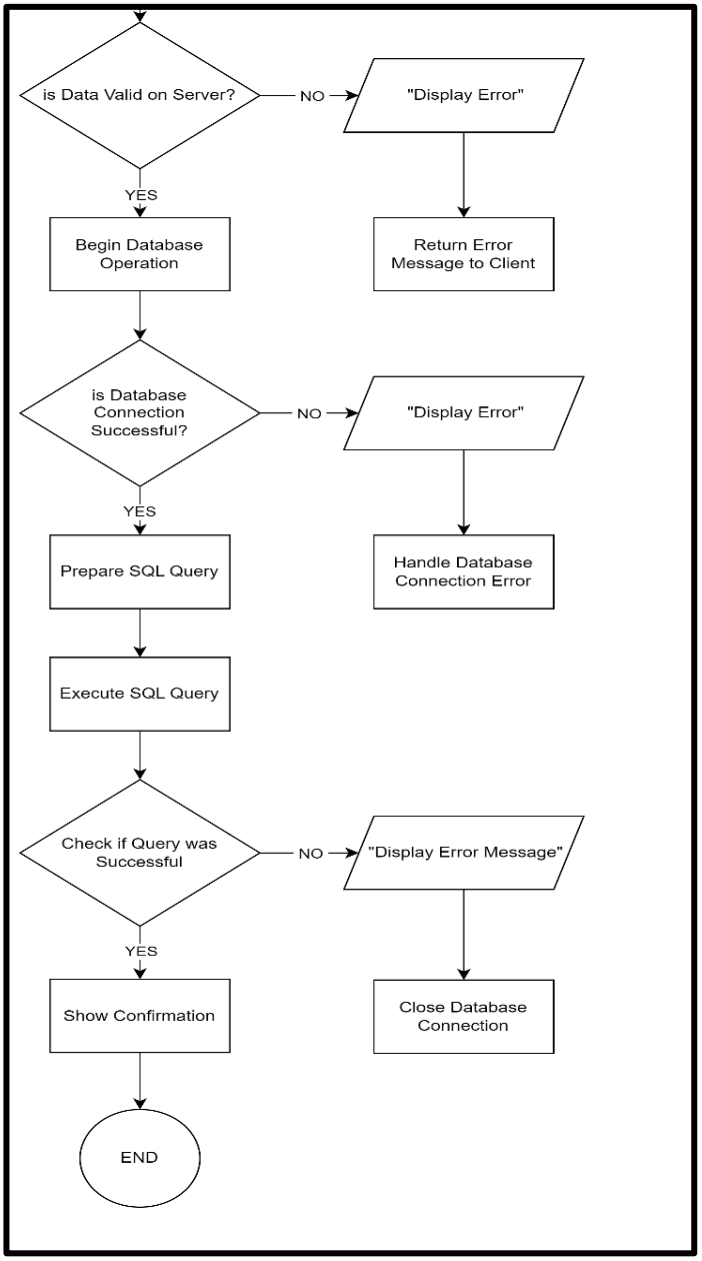

FITRIYAH IML208
nurulfitriyah.py
nurulfitriyah.py > collect_data
for widget in student_info_frame.winfo_children():
widget.grid_configure(padx=60, pady=5)
# Frame 2
parent_info_frame = tkinter.LabelFrame(frame, bg= "#BBA4F4", text="Father/ Mother/ Guardian Information")
parent_info_frame.grid(row=0, column=1, padx=20, pady=0)
#Parents Full Name
parent_full_name = tkinter.Label(parent_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Full Name")
parent_full_name.grid(row=0, column=0)
parent_full_name_entry = tkinter.Entry(parent_info_frame)
parent_full_name_entry.grid(row=1, column=0)
#Parent Email Info
parent_email = tkinter.Label(parent_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Email")
parent_email.grid(row=0, column=1)
parent_email_entry = tkinter.Entry(parent_info_frame)
parent_email_entry.grid(row=1, column=1)
for widget in parent_info_frame.winfo_children():
widget.grid_configure(padx=60, pady=20)
#Frame 3
stat_pack_frame =tkinter.LabelFrame(frame, bg= "#BBA4F4", text="Music Class Package")
stat_pack_frame.grid(row= 1, column=0, padx=20, pady=10)
#Prices list using textbox
prices_text = tkinter.Text(stat_pack_frame, bg="#E9C3E1", fg='#31572c', height=15, width=45, font= ("Cooper Black", 10))
#Define list by using pricebox
prices_text.insert(tkinter.END, "Package 1:\n Beginner Class,\nFree Food\nRM 80\n\n")
prices_text.insert(tkinter.END, "Package 2:\n Intermediate Class\nFree Food\nRM 150\n\n")
Ln 62, Col 24
Spaces: 4
UTF-8
CRLF
Python
3.11.7 64-bit (Microsoft Store)


FITRIYAH IML208
nurulfitriyah.py
nurulfitriyah.py > collect_data
prices_text.insert(tkinter.END, "Package 2:\n Intermediate Class\nFree Food\nRM 150\n\n")
prices_text.insert(tkinter.END, "Package 3:\n Deluxe Class\nFree Food\nRM200\n\n")
prices_text.configure(state='disabled')
prices_text.grid(padx=70, pady=0)
#----------
#Frame 4
sp_info_frame = tkinter.LabelFrame(frame, bg= "#BBA4F4", text="Select & Pay")
sp_info_frame.grid(row= 1, column=1, padx=20, pady=0)
#Set
student_set = tkinter.Label (sp_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Package")
student_set.grid(row=0, column=0)
student_set_combobox= ttk.Combobox (sp_info_frame, values=["Package 1", "Package 2", "Package 3"])
student_set_combobox.grid(row=0, column=1)
#Quantity
student_pack_quantity = tkinter.Label(sp_info_frame, bg= "#E9C3E1", width=10, font= ("Times New Roman", 10), text="Quantity / Pax")
student_pack_quantity.grid(row=1, column=0)
student_pack_quantity_entry = tkinter.Entry(sp_info_frame)
student_pack_quantity_entry.grid(row=1, column=1)
for widget in sp_info_frame.winfo_children():
widget.grid_configure(padx=70, pady=45)
#----------
# Button
save_button = tkinter.Button(root, bg= "#926EED", text="Submit Data", font= ("Times New Roman", 15), command=collect_data)
save_button.pack(pady=5)
# Print Output & result
label = tkinter.Label(root, bg= "#926EED", text='Payment Details:', font=("Times New Roman",15))
label.pack(ipadx=10, ipady=10)
output_label = tkinter.Label(root, text="")
output_label.pack()
Ln 62, Col 24
Spaces: 4
UTF-8


save_button = tkinter.Button(root, bg= "#926EED", text="Submit Data", font= ("Times New Roman", 15), command=collect_data)
save_button.pack(pady=5)
# Print Output & result
label = tkinter.Label(root, bg= "#926EED", text='Payment Details:', font=("Times New Roman",15))
label.pack(ipadx=10, ipady=10)
output_label = tkinter.Label(root, text="")
output_label.pack()
root.mainloop()

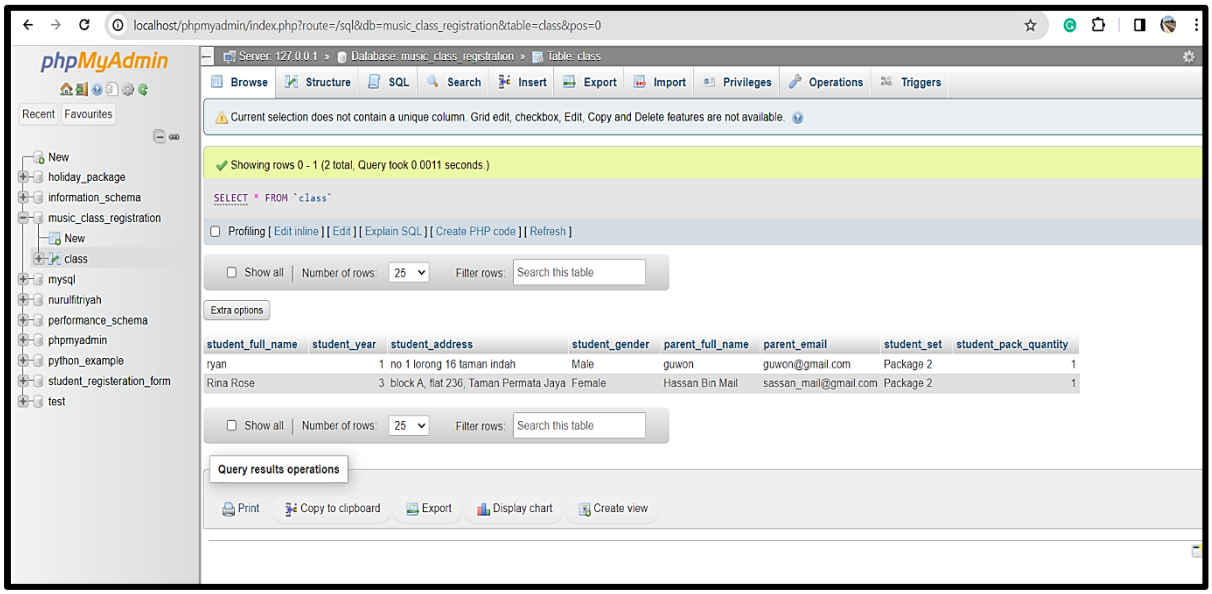
3.0 FLOWCHART

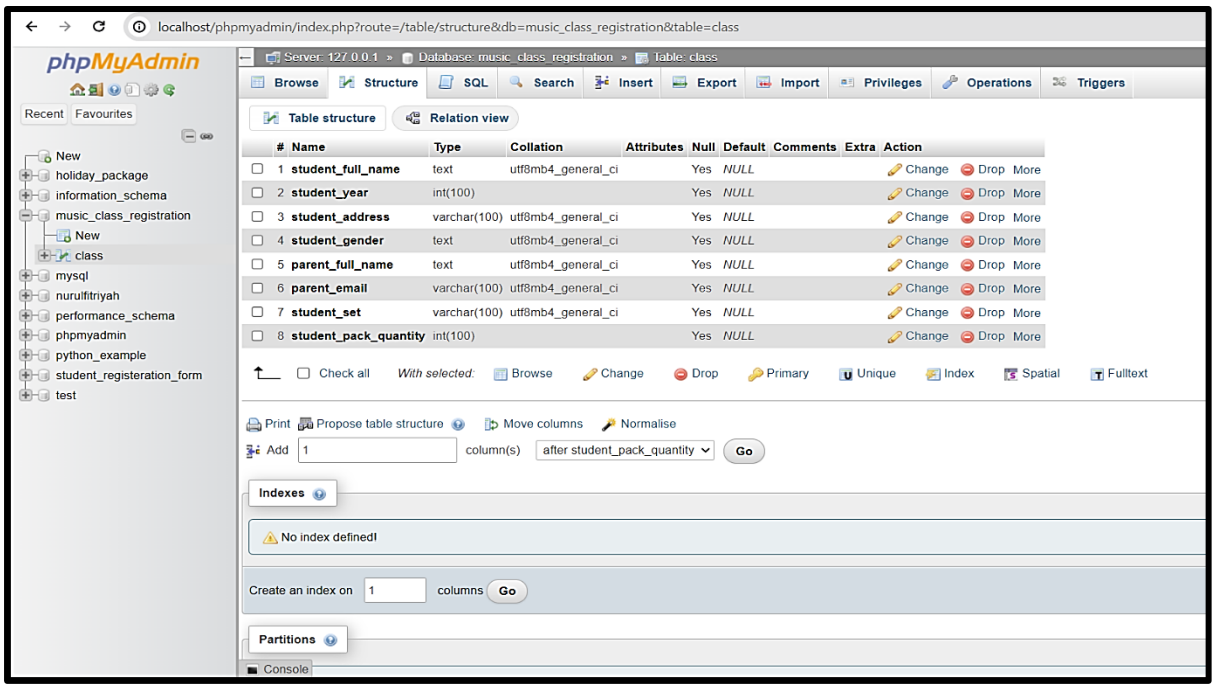




3.0 GRAPHICAL USER INTERFACE (GUI)

4.0 DATABASE


localhost/phpmyadmin/index.php?route=/sql&db=music_class_registration&table=class&pos=0
phpMyAdmin
Server: 127.0.0.1 » Database: music_class_registration » Table: class
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
Recent Favourites
New
holiday_package
information_schema
music_class_registration
New
class
mysql
nurulfitriyah
performance_schema
phpmyadmin
python_example
student_registeration_form
test
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Showing rows 0 - 1 (2 total, Query took 0.0011 seconds.)
SELECT * FROM `class`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
Extra options
student_full_name student_year student_address student_gender parent_full_name parent_email student_set student_pack_quantity
ryan 1 no 1 lorong 16 taman indah Male guwon guwon@gmail.com Package 2 1
Rina Rose 3 block A, flat 236, Taman Permata Jaya Female Hassan Bin Mail sassan_mail@gmail.com Package 2 1
Show all | Number of rows: 25 Filter rows: Search this table
Query results operations
Print Copy to clipboard Export Display chart Create view


localhost/phpmyadmin/index.php?route=/table/structure&db=music_class_registration&table=class
phpMyAdmin
Server: 127.0.0.1 » Database: music_class_registration » Table: class
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
Recent Favourites
New
holiday_package
information_schema
music_class_registration
New
class
mysql
nurulfitriyah
performance_schema
phpmyadmin
python_example
student_registeration_form
test
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 student_full_name text utf8mb4_general_ci Yes NULL Change Drop More
2 student_year int(100) Yes NULL Change Drop More
3 student_address varchar(100) utf8mb4_general_ci Yes NULL Change Drop More
4 student_gender text utf8mb4_general_ci Yes NULL Change Drop More
5 parent_full_name text utf8mb4_general_ci Yes NULL Change Drop More
6 parent_email varchar(100) utf8mb4_general_ci Yes NULL Change Drop More
7 student_set varchar(100) utf8mb4_general_ci Yes NULL Change Drop More
8 student_pack_quantity int(100) Yes NULL Change Drop More
Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext
Print Propose table structure Move columns Normalise
Add 1 column(s) after student_pack_quantity Go
Indexes
No index defined!
Create an index on 1 columns Go
Partitions
Console