UNIVERSITI TEKNOLOGI MARA

COLLEGE OF COMPUTING, INFORMATICS AND MATHEMATICS,

UNIVERSITI TEKNOLOGI MARA,

MERBOK, KEDAH

DIPLOMA IN LIBRARY INFORMATICS

(CDIM144)

PROGRAMMING FOR LIBRARIES

(IML208)

“INDIVIDUAL PROJECT”

PREPARED BY:

WAN NUR ALLIYA BINTI WAN MOHD KHAIRUDDIN (2022873652)

CLASS: KCDIM1443F

PREPARED FOR:

SIR AIRUL SHAZWAN BIN NORSHAHIMI

SUBMISSION DATE:

WEEK 12

“INDIVIDUAL PROJECT”

PREPARED BY:

WAN NUR ALLIYA BINTI WAN MOHD KHAIRUDDIN (2022873652)

COLLEGE OF COMPUTING, INFORMATICS AND MATHEMATICS,

UNIVERSITI TEKNOLOGI MARA,

MERBOK, KEDAH

ACKNOWLEDGEMENT

I would like to take this opportunity to extend my sincere appreciation to several individuals who have played a crucial role in the successful completion of this assignment.

First and foremost, I am immensely grateful to Sir Airul Shazwan Bin Norshahimi for their unwavering support, invaluable insights, and dedication to fostering a deeper understanding of the subject matter. Their guidance and mentorship have been instrumental in shaping the quality and depth of this assignment.

Last but not least, I wish to thank my family and friends for their understanding and encouragement during this academic journey. Their unwavering support provided me with the motivation and resilience necessary to complete this assignment effectively.

Table of contents

1.0 INTRODUCTION

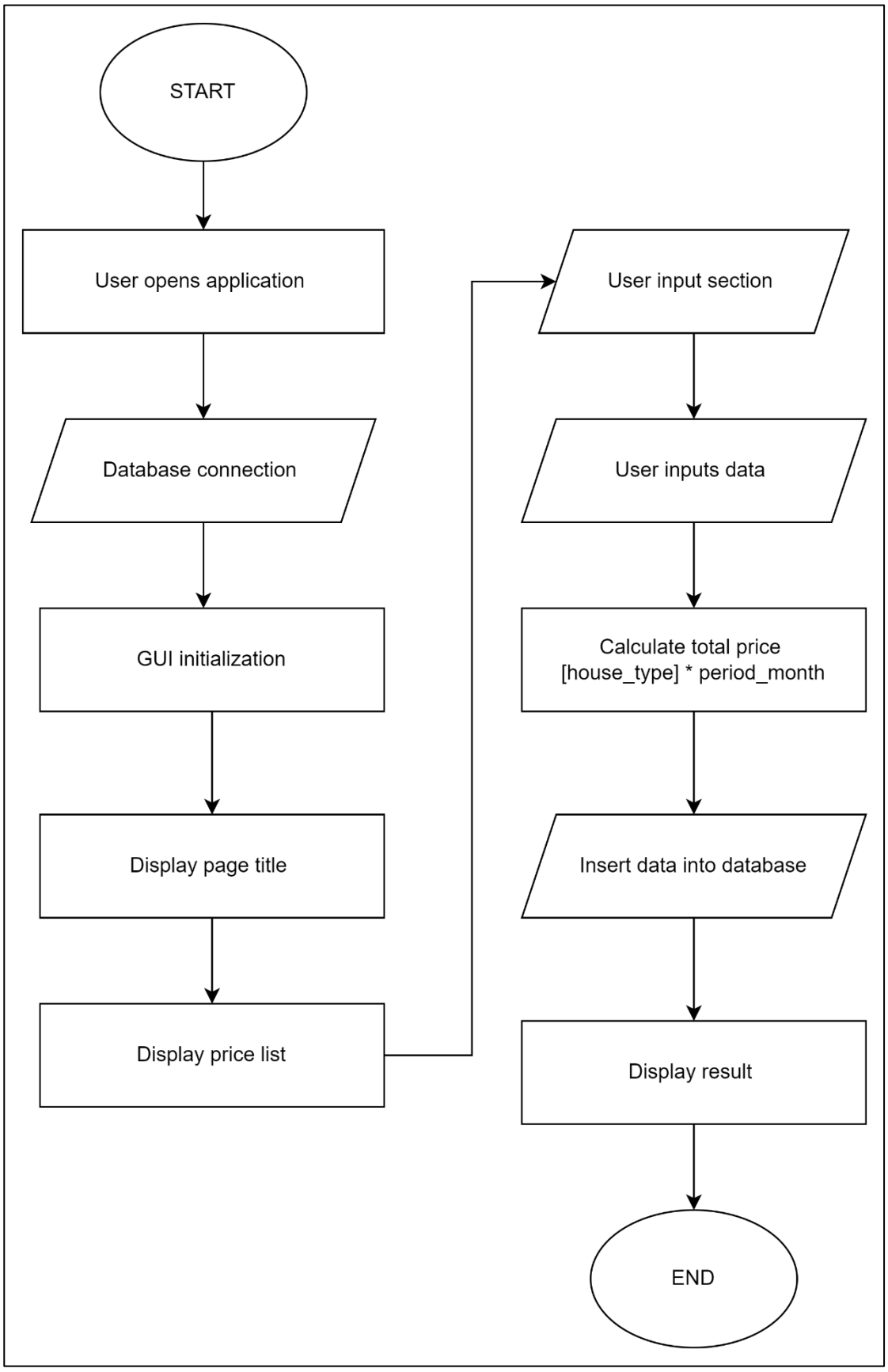
The given A straightforward GUI application written in Python that calculates house rentals and stores data in a MySQL database is called a Tkinter library application. Users of the application can choose a house type, a month-by-month rental duration, and a house location from drop-down menus. When the user clicks the "Calculate" button, the application calculates the total rental cost based on the options they have chosen and adds the necessary information to a MySQL database table called `user}. The application offers an easy-to-use interface for entering rental information and seeing the figures that are generated.

Functionality and Features:

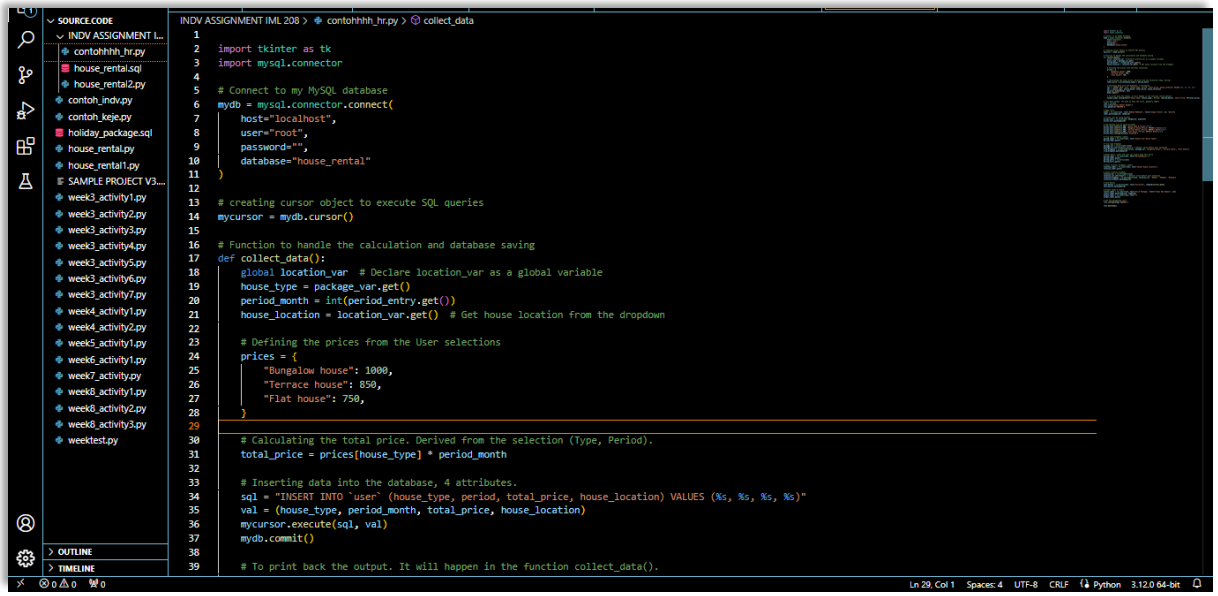
Interface: The GUI application offers a user-friendly interface with dropdown menus for location and house type, a field for entering the rental period, and a "Calculate" button that starts the calculation and saves the data.

- Price List Display: To assist users in making educated decisions, a text box shows the costs for various house types.
- Data Calculation: Using a predefined dictionary of prices, the script determines the total rental price based on the chosen house type and rental period.
- Database Interaction: After establishing a connection with a MySQL database, the program inserts user input into the user table and creates a cursor object.
- Output Display: The application shows the selected options and the computed total price following the calculation and database insertion.

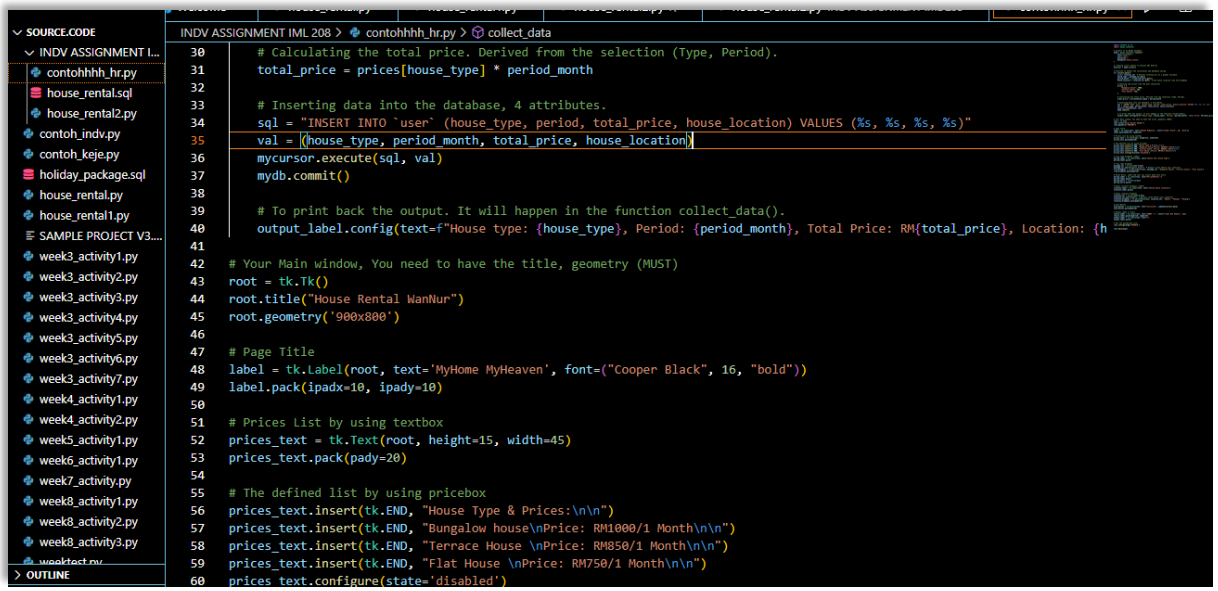
2.0 FLOWCHART



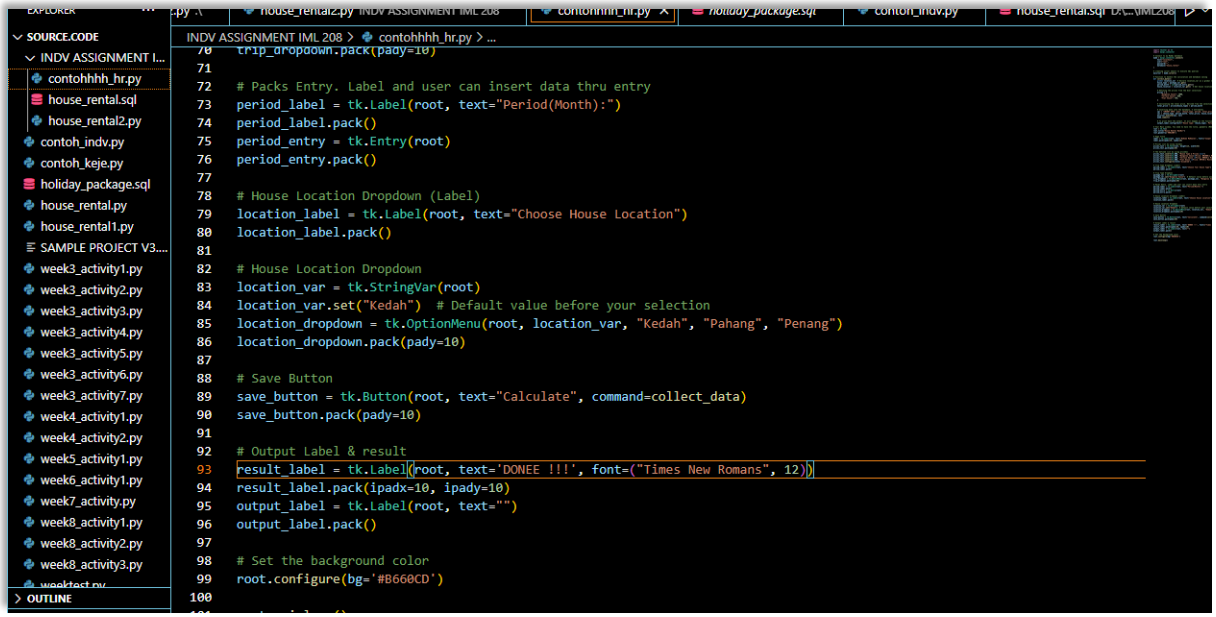
3.0 SNAPSHOT OF THE CODE



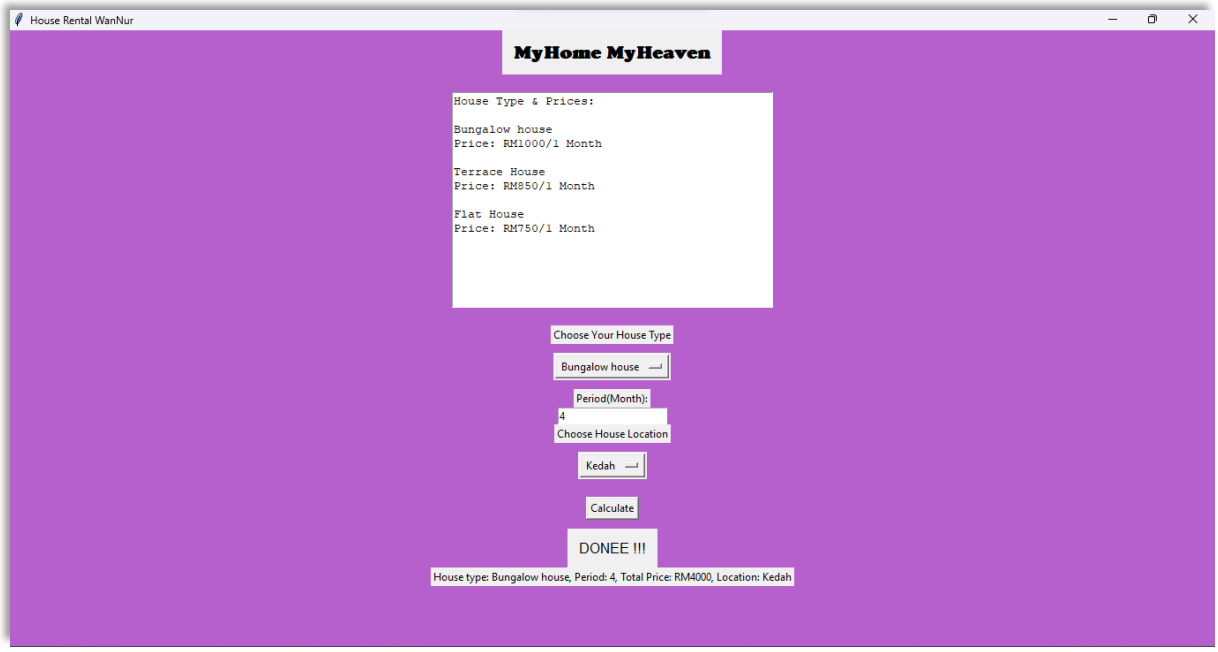
```python
import tkinter as tk
import mysql.connector

# Connect to my MySQL database
mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="house_rental"
)

# creating cursor object to execute SQL queries
mycursor = mydb.cursor()

# Function to handle the calculation and database saving
def collect_data():
    global location_var  # Declare location_var as a global variable
    house_type = package_var.get()
    period_month = int(period_entry.get())
    house_location = location_var.get()  # Get house location from the dropdown

    # Defining the prices from the User selections
    prices = {
        "Bungalow house": 1000,
        "Terrace house": 850,
        "Flat house": 750,
    }

    # Calculating the total price. Derived from the selection (Type, Period).
    total_price = prices[house_type] * period_month

    # Inserting data into the database, 4 attributes.
    sql = "INSERT INTO `user` (house_type, period, total_price, house_location) VALUES (%s, %s, %s, %s)"
    val = (house_type, period_month, total_price, house_location)
    mycursor.execute(sql, val)
    mydb.commit()

    # To print back the output. It will happen in the function collect_data().
```



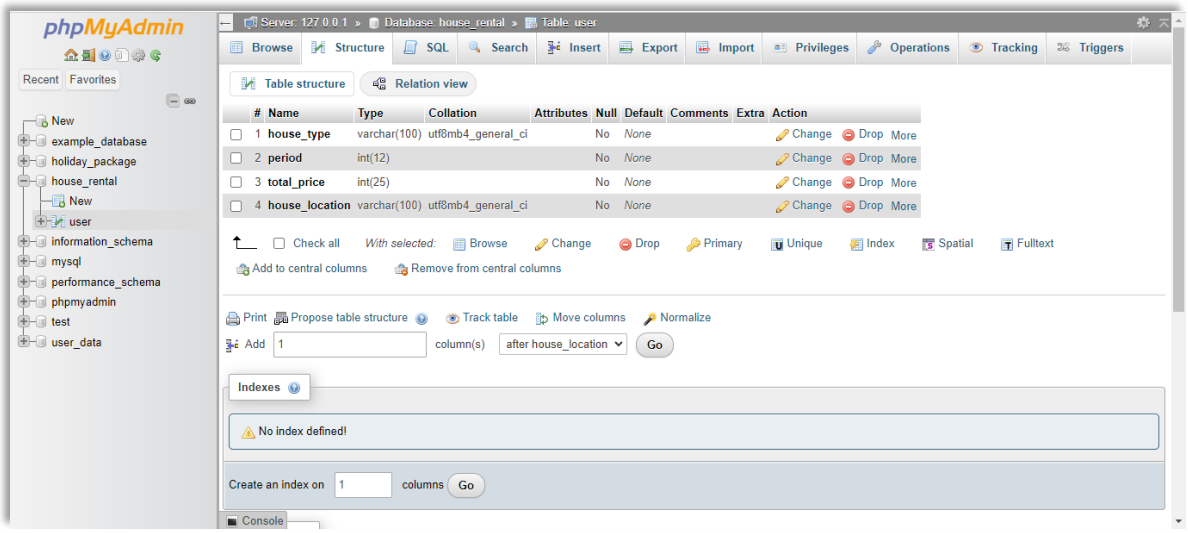
```python
    # Calculating the total price. Derived from the selection (Type, Period).
    total_price = prices[house_type] * period_month

    # Inserting data into the database, 4 attributes.
    sql = "INSERT INTO `user` (house_type, period, total_price, house_location) VALUES (%s, %s, %s, %s)"
    val = (house_type, period_month, total_price, house_location)
    mycursor.execute(sql, val)
    mydb.commit()

    # To print back the output. It will happen in the function collect_data().
    output_label.config(text=f"House type: {house_type}, Period: {period_month}, Total Price: RM{total_price}, Location: {h

# Your Main window, You need to have the title, geometry (MUST)
root = tk.Tk()
root.title("House Rental WanNur")
root.geometry('900x800')

# Page Title
label = tk.Label(root, text='MyHome MyHeaven', font=("Cooper Black", 16, "bold"))
label.pack(ipadx=10, ipady=10)

# Prices List by using textbox
prices_text = tk.Text(root, height=15, width=45)
prices_text.pack(pady=20)

# The defined list by using pricebox
prices_text.insert(tk.END, "House Type & Prices:\n\n")
prices_text.insert(tk.END, "Bungalow house\nPrice: RM1000/1 Month\n\n")
prices_text.insert(tk.END, "Terrace House \nPrice: RM850/1 Month\n\n")
prices_text.insert(tk.END, "Flat House \nPrice: RM750/1 Month\n\n")
prices_text.configure(state='disabled')
```

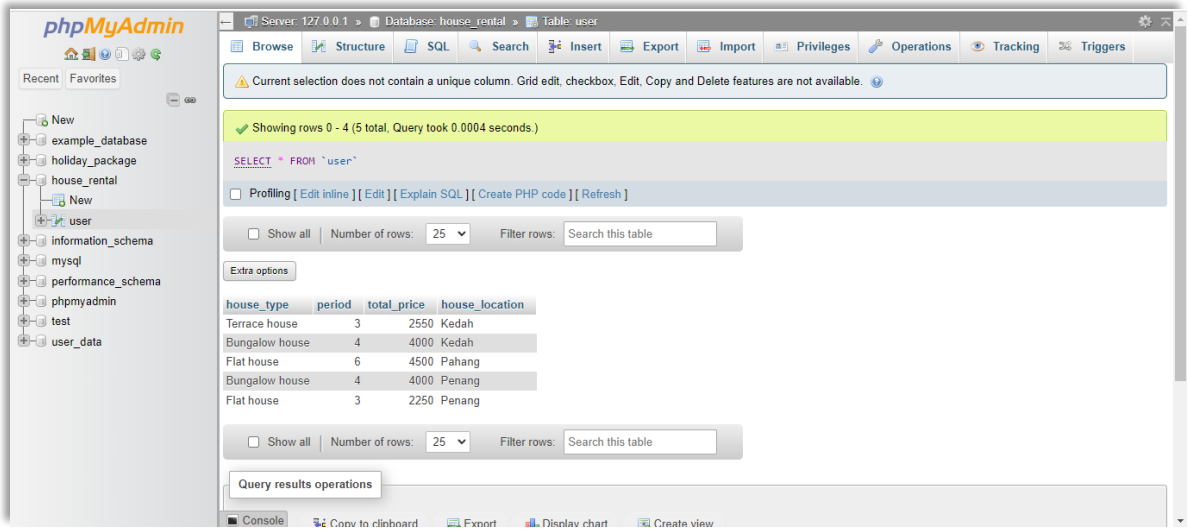

SOURCE.CODE
INDV ASSIGNMENT I...
contohhhh_hr.py
house_rental.sql
house_rental2.py
contoh_indv.py
contoh_keje.py
holiday_package.sql
house_rental.py
house_rental1.py
SAMPLE PROJECT V3....
week3_activity1.py
week3_activity2.py
week3_activity3.py
week3_activity4.py
week3_activity5.py
week3_activity6.py
week3_activity7.py
week4_activity1.py
week4_activity2.py
week5_activity1.py
week6_activity1.py
week7_activity.py
week8_activity1.py
week8_activity2.py
week8_activity3.py
OUTLINE
INDV ASSIGNMENT IML 208 > contohhhh_hr.py > ...
70 trip_dropdown.pack(pady=10)
71
72 # Packs Entry. Label and user can insert data thru entry
73 period_label = tk.Label(root, text="Period(Month):")
74 period_label.pack()
75 period_entry = tk.Entry(root)
76 period_entry.pack()
77
78 # House Location Dropdown (Label)
79 location_label = tk.Label(root, text="Choose House Location")
80 location_label.pack()
81
82 # House Location Dropdown
83 location_var = tk.StringVar(root)
84 location_var.set("Kedah") # Default value before your selection
85 location_dropdown = tk.OptionMenu(root, location_var, "Kedah", "Pahang", "Penang")
86 location_dropdown.pack(pady=10)
87
88 # Save Button
89 save_button = tk.Button(root, text="Calculate", command=collect_data)
90 save_button.pack(pady=10)
91
92 # Output Label & result
93 result_label = tk.Label(root, text='DONEE !!!', font=("Times New Romans", 12))
94 result_label.pack(ipadx=10, ipady=10)
95 output_label = tk.Label(root, text="")
96 output_label.pack()
97
98 # Set the background color
99 root.configure(bg='#B660CD')
100

4.0 SNAPSHOT OF GUI



5.0 SNAPSHOT OF DATABASE





6.0 CONCLUSION

In conclusion, this Python script builds a simple house rental application by utilizing the MySQL connector for database interaction and the Tkinter library for creating a graphical user interface. It's simple for users to enter their preferences, get the total cost of the rental, and see the outcomes. For a more comprehensive home rental management system, this script can be extended to include more features, validations, and enhancements to the database interactions and user interface.