UNIVERSITI TEKNOLOGI MARA (UITM), CAWANGAN KEDAH, KAMPUS SUNGAI PETANI

COLLAGE OF COMPUTING, INFORMATIC AND MATHEMATIC

DIPLOMA IN LIBRARY INFORMATIC

(CDIM144)

PROGRAMMING FOR LIBRARIES

(IML208)

ASSINGMENT I: INDIVIDUAL ASSIGMENT- REPORT ON SIMPLE COMPUTER INTERFACE (LIBRARY ORGANIZER)

PREPARED BY:

MUHAMMAD DANISH ALFIAN BIN MOHD ZULLKIFLI

2022625348

PREPARED FOR:

Mr. AIRUL SHAZWAN BIN NORSHAHIMI

SUBMISSION DATE: 4TH JANUARY 2024

ASSINGMENT I: INDIVIDUAL ASSIGMENT- REPORT ON SIMPLE COMPUTER INTERFACE (LIBRARY ORGANIZER)

MUHAMMAD DANISH ALFIAN BIN MOHD ZULLKIFLI

2022625348

DIPLOMA IN LIBRARY INFORMATIC

COLLAGE OF COMPUTING, INFORMATIC AND MATHEMATIC

UNIVERSITI TEKNOLOGI MARA (UITM), CAWANGAN KEDAH

SUBMISSION DATE: 4TH JANUARY 2024

STUDENT PLEDGE OF ACADEMIC INTEGRITY

As a student of Universiti Teknologi MARA (UiTM), it is my responsibility to act in accordance with UiTM's academic assessment and evaluation policy. I hereby pledge to act and uphold academic integrity and pursue scholarly activities in UiTM with honesty and responsible manner. I will not engage or tolerate acts of academic dishonesty, academic misconduct, or academic fraud including but not limited to:

a. **Cheating:** Using or attempt to use any unauthorized device, assistance, sources, practice or materials while completing academic assessments. This include but not limited to copying from another, allowing others to copy, unauthorized collaboration on an assignment or open book tests, or engaging in any act or conduct that can be construed as cheating.
b. **Plagiarism:** Using or attempts to use the work of others (ideas, design, words, art, music, etc.) without acknowledging the source; using or purchasing materials prepared by another person or agency or engaging in other behavior that a reasonable person would consider as plagiarism.
c. **Fabrication:** Falsifying data, information, or citations in any academic assessment and evaluation.
d. **Deception:** Providing false information with intend to deceive an instructor concerning any academic assessment and evaluation.
e. **Furnishing false information:** Providing false information or false representation to any UiTM official, instructor, or office.

With this pledge, I am fully aware that I am obliged to conduct myself with utmost honesty and integrity. I fully understand that a disciplinary action can be taken against me if I, in any manner, violate this pledge.

Name : MUHAMMAD DANISH ALFIAN BIN MOHD ZULLKIFLI
Matric Number : 2022625348
Course Code : IML208
Programme Code :-
Faculty / Campus : UiTM Kampus Sungai Petani

ACKNOWLEDGEMENT

Assalamulaikum w.b.t

First and foremost, I would like to praise and thank the almighty god for providing me with the strength, time and patience. As a result of his blessing, I was able to completed his assignment flourishingly. Without his blessing, I would not get this far. With all of my full commitment and responsibility, I will produce a good assignment with all of my strength.

Secondly, I would like to express my gratitude to the lecturer that helped me in this assignment, Mr. Airul Shazwan Bin Norshahimi, because this assignment would not have been completed properly without his guidance. He always there to support and guide me on how to complete this assignment with full of patience and clear instruction to produce good results. He was a huge inspiration to me as i worked on this project. I would also like to thank him for teaching me this course.

Last but not lease, I would like to show thanks to my loving family for their encouragement and prayers that have kept me going till now. They have always been there for me, give us a call lending their ears for me to keep me going up. I hope my effort was worth it and will help me to get good grade for my final results. After doing this assignment I have learned new things that I never knew before. May this assignment will provide me something and help me in the future.

Table Of Contents

Contents

1.0 INTRODUCTION

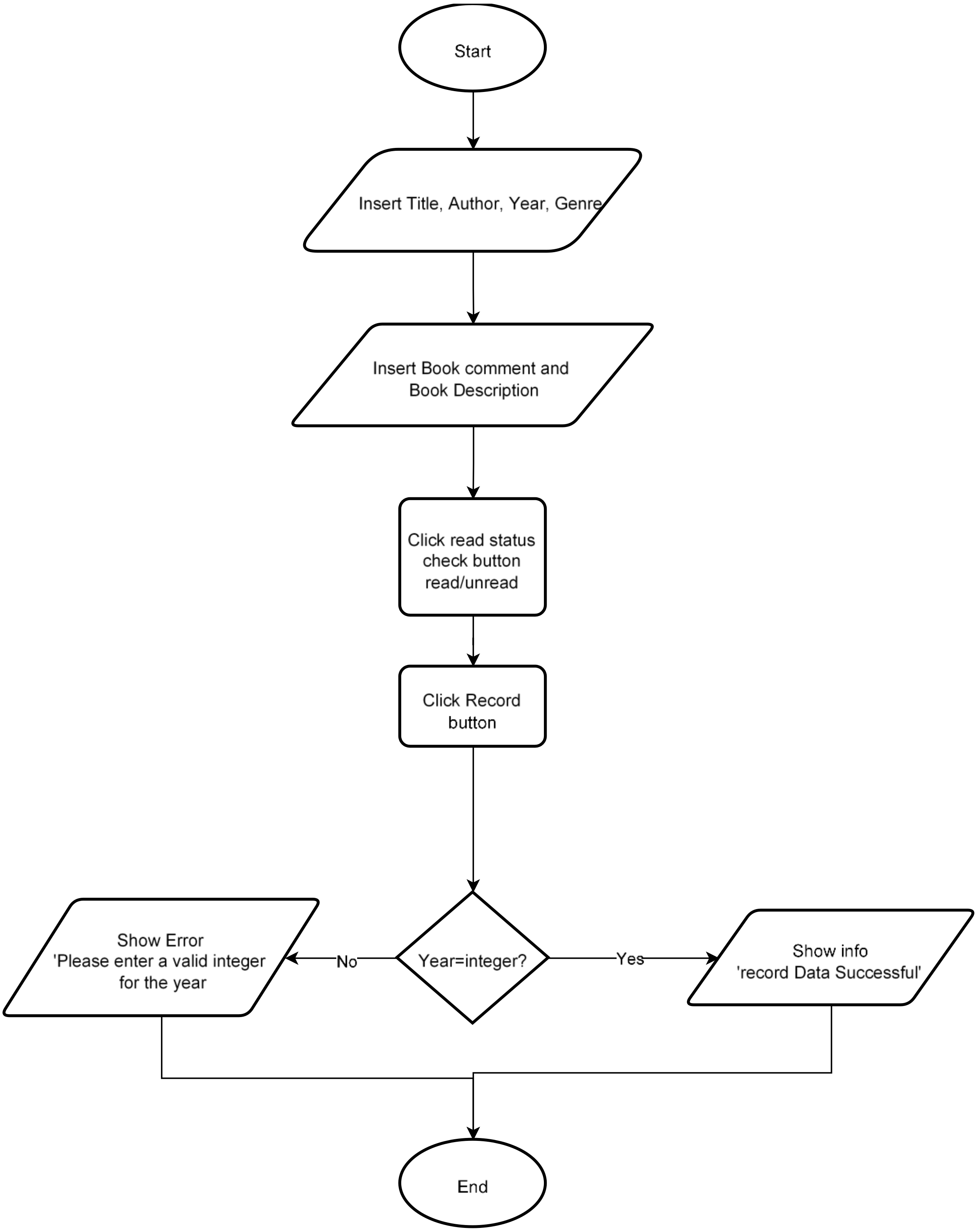
In this individual assignment, the simple computer interface that I choose to make is called 'library personal organizer'. The reason that I choose to make this library organizer is because I'm inspired to do it when I first watch this suggestion in the google, and the reason of that is because I dream that when I have my own house, I would want my personal library so without a doubt that I would need a library organizer for that as I'm library student learned to arrange books systematically and I want to practice that in real life. Basically, the system library organizer will record all your book data to the database so you can keep track on all of your books. Other than that, it will record the information of a person who want to loan it so you would know who loan your books at the moment.

Now, you might wonder what does the data that user might insert into this system to record it in the database, firstly to record your book you need to insert your book title, author, year, then genre after that it optional for you to insert book comment and book description. After that you have to click on read or unread check button and lastly click on 'enter data' button to record your data. After you record your data, you can see your book record on the book list frame below.

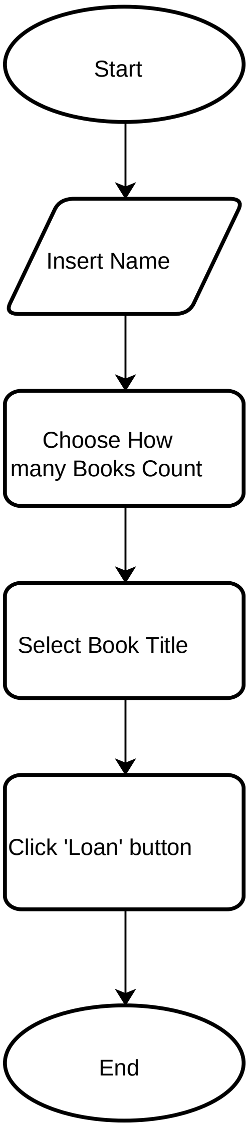
Then if there are people want to loan your books, they need to put a simple information to record, such as name, numbers off books that he wants to loan and the books name. after that you have to click 'loan button' to record it and it also will automatically count how much you have to pay based on how many books you loan. However, there are some rules to loan books such as one book equal to RM 0.50, maximum loan at one time only 5, then books must be return after 3 weeks and lastly if the is loss the borrower need to pay full price of the books.

2.0 FLOWCHART

2.1 Enter Data Flowchart


Start
Insert Title, Author, Year, Genre
Insert Book comment and Book Description
Click read status check button read/unread
Click Record button
Year=integer?
No
Yes
Show Error 'Please enter a valid integer for the year
Show info 'record Data Successful'
End

2.2 Book Loan Flowchart



3.0 PYTHON TKINTER CODE

```
import tkinter as tk
import array
from tkinter import ttk
from tkinter import messagebox
from tkinter import font
import os
#import pandas as pd
#df=pd.read_excel("C:\Users\user\Documents\source_code\colours.xlsx") # Path of the file.
#my_list=df['Color Name'].values.tolist()

#from tkinter import PhotoImage
import mysql.connector


def refresh_window():
    # Redraw the window

    os.popen("assingment2.py")
    #print("Refresh completed.")
    root.destroy()



def update_book_list():
    mycursor.execute("SELECT * FROM book")
    books = mycursor.fetchall()


    # Clear existing items in the listbox
    book_list.delete(0, tk.END)

    # Insert fetched data into the listbox
    for book in books:
        book_list.insert(tk.END, book)

# Create a cursor object to execute SQL queries

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="book_entry"
)

mycursor = mydb.cursor()
```

```
def enter_data():



    title_value = title_entry.get()
    author_value = author_entry.get()
    year_value = (year_entry.get())
    genre_value = genre_combobox.get()
    comment_value = comment_entry.get("1.0", tk.END)  # Use get("1.0", tk.END) for Text widget
    book_desc_value = book_desc_entry.get("1.0", tk.END)
    read_status_value = read_status_var.get()


    sql= 'INSERT INTO book (title, author, genre, year, comment, description,  status ) VALUES (%s, %s, %s, %s, %s, %s, %s)'
    val= (title_value, author_value, genre_value, year_value, comment_value, book_desc_value, read_status_value)

    mycursor.execute(sql, val)
    mydb.commit()

    if not year_value.isdigit():

        messagebox.showerror("Error", "Please enter a valid integer for the year.")
    else:

        update_book_list()
        entered_data = f"TITLE: {title_value}\nAUTHOR: {author_value}\nGENRE: {genre_value}\nYEAR: {year_value}\nCOMMENT: {comment_value}\nBOOK DESCRIPTION : {book_desc_value}\nREAD STATUS: {read_status_value}\n"
        entry_data.config(state='normal')
        entry_data.delete(1.0, tk.END)
        entry_data.insert(tk.END, entered_data)
        entry_data.config(state='disabled')
        messagebox.showinfo('Sucess', 'Record Data Sucessful')


    #entered_data = f"TITLE: {title_value}\nAUTHOR: {author_value}\nGENRE: {genre_value}\nYEAR: {year_value}\nCOMMENT: {comment_value}\nBOOK DESCRIPTION : {book_desc_value}\nREAD STATUS: {read_status_value}\n"
    #entry_data.config(state='normal')
    #entry_data.delete(1.0, tk.END)
    #entry_data.insert(tk.END, entered_data)
    #entry_data.config(state='disabled')
    try:

        # Show success message
        pass



    except mysql.connector.Error as err:
        # Handle errors
        print(f"Error: {err}")
        messagebox.showerror("Error", f"Error: {err}")



def comboclick(event) :
    book_loan.insert(tk.END, book_entry.get())


def loan():
    book_count = float(count_spinbox.get())
    price=book_count*0.50


    loan_data= f'NAME: {name_entry.get()}\nPRICE(RM): {price:.2f}\nBOOK:
{book_loan.get("1.0", tk.END)}'
    loan_entry.config(state='normal')
    loan_entry.delete(1.0, tk.END)  # Clear existing content
    loan_entry.insert(tk.END, loan_data)
    loan_entry.config(state='disabled')

    namer=name_entry.get()
    counts=int(book_count)
    namebook=book_loan.get('1.0', tk.END)

    sql= 'INSERT INTO loan (name, count, book, price) VALUES (%s, %s, %s, %s
)'
    val= (namer, counts, namebook, price)

    mycursor.execute(sql, val)
    mydb.commit()
```

```
def on_configure(event):
    canvas.configure(scrollregion=canvas.bbox('all'))

root = tk.Tk()
root.title("Library Organizer")
root.geometry("709x600")

# main frame
main_frame= tk.Frame(root)
main_frame.pack(fill='both', expand=1)

#root.resizable(False, False)


#buat canvas
canvas=tk.Canvas(main_frame, scrollregion= (0,0,2000,5000), )
canvas.pack(side='left', fill='both', expand=True)
canvas.bind('<MouseWheel>',  lambda event: canvas.yview_scroll(-
int(event.delta / 60), 'units'))

#buat scrollbar
#scrollbar=tk.Scrollbar(main_frame, orient='vertical', command=canvas.yview)
scrollbar=ttk.Scrollbar(root, orient='vertical', command=canvas.yview)
scrollbar.pack(side='right', fill='y')
canvas.configure(yscrollcommand=scrollbar.set)
canvas.bind('<MouseWheel>',  lambda event: canvas.yview_scroll(-
int(event.delta / 60), 'units'))
canvas.bind ('<Configure>', on_configure)
canvas.configure(yscrollcommand=scrollbar.set)

scrollbar.place(relx=1,rely=0, relheight=1, anchor='ne')

frame_container = tk.Frame(canvas, bg="white", padx=30)
canvas.create_window((0, 0), window=frame_container, anchor='nw')

#input frame inside canvas
frame2=tk.Frame(frame_container, bg="#4D4637")


#canvas.create_window((0,0), window=frame2, anchor='nw' )
frame2.grid(row=0,column=0)
#frame2.pack()
#frame2.grid(row=0, column=0)

label = tk.Label(frame2, text="Click the below button to refresh the window.",
bg="#4D4637", fg='white')
label.grid(row=0, column=0)
```

```
button = tk.Button(frame2, text="Refresh", command=refresh_window,
bg="#4D4637", fg='white')
button.grid(row=1, column=0)

#title label
label_title= tk.Label(frame2, text="MY PERSONAL LIBRARY \nORGANIZER", font= (
"Arial Black", 30, ) , fg= 'white' , bg="#4D4637", pady=50, padx=40 )
label_title.grid(row=2, column=0 )

frame3=tk.Frame(frame_container, padx=20, pady=30, bg="#FFDE82" )


#canvas.create_window((0,0), window=frame3, anchor='nw' )
frame3.grid(row=1,column=0, pady=30)
#frame3.pack()
#book entry frame
entry_frame= tk.LabelFrame(frame3, text="BOOK DATA ENTRY", pady=30, padx=25,
font= ( "Arial Black",  ), bg="#FFDE82")
entry_frame.grid(row=0 , column=0, padx=20, pady=10)

title_label=tk.Label(entry_frame, text='Title', font=( 'Bahnschrift',
12),   bg='#FFDE82')
title_label.grid(row=0, column=0)

author_label=tk.Label(entry_frame, text='Author', font=( 'Bahnschrift',
12),bg='#FFDE82')
author_label.grid(row=0, column=1)

title_entry=tk.Entry(entry_frame,  bg='#FFCB97', )
author_entry=tk.Entry(entry_frame, bg='#FFCB97')
title_entry.grid(row=1, column=0)
author_entry.grid(row=1, column=1)



genre_label=tk.Label(entry_frame, text='Genre',font=( 'Bahnschrift', 12),
bg='#FFDE82')
genre_combobox=ttk.Combobox(entry_frame,
values=['Fiction','Novel','Narrative','Mystery','History','Short
Story','Horror','Philosophy','Science','Biology','Spirituality','Poetry','Comi
c','Language','Essay'],)
genre_label.grid(row=3, column=1)
genre_combobox.grid(row=4, column=1)
#genre_combobox.set('select genre')
```

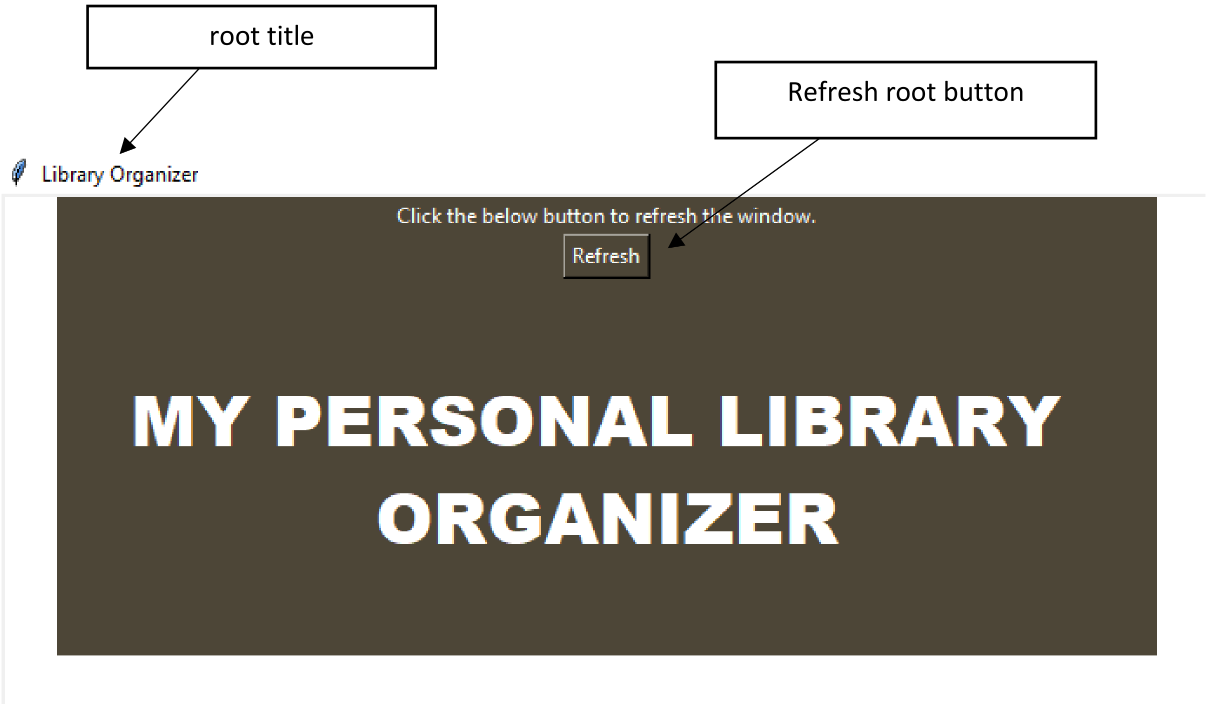
```
year_label=tk.Label(entry_frame, text='Year',font=( 'Bahnschrift', 12),
bg='#FFDE82')
year_entry=tk.Entry(entry_frame, bg='#FFCB97')
year_label.grid(row=3, column=0)
year_entry.grid(row=4,column=0)

comment_label=tk.Label(entry_frame, text='Book Comment', font=( 'Bahnschrift',
12),bg='#FFDE82')
comment_entry=tk.Text(entry_frame, height=4, width=25, bg='#FFCB97')
comment_label.grid(row=6, column=0)
comment_entry.grid(row=7,column=0)


book_desc=tk.Label(entry_frame, text='Book Description',font=( 'Bahnschrift',
12), bg='#FFDE82')
book_desc_entry=tk.Text(entry_frame, height=4, width=25, bg='#FFCB97')
book_desc.grid(row=6, column=1)
book_desc_entry.grid(row=7, column=1)

read_status_var=tk.StringVar()
read_button=tk.Checkbutton(entry_frame, text='Read',font=( 'Bahnschrift', 12),
variable=read_status_var, onvalue='Read' , offvalue='Unread', bg='#FFDE82')
read_button.grid(row=8, column=0, sticky='ew')




#button

button=tk.Button(entry_frame, text='Enter Data', pady=5, bg='#FFC3AE', font=(
'Bahnschrift'), command=enter_data)#command= enter_data )
button.grid(row=8, column=1, sticky='ew', )

entry_data=tk.Text(entry_frame, height=10, width=35, bg='#FFC3AE' )
entry_data.grid(row=9, column=0, columnspan=2 , )




for widget in entry_frame.winfo_children():
    widget.grid_configure(padx=10, pady=5)
    button.grid_configure(padx=30, pady=30)




#book list frame

frame4=tk.Frame(frame_container, padx=101, pady=30, bg='#CAA74E')

```

```
#canvas.create_window((0,0), window=frame4, anchor='nw' )
frame4.grid(row=2,column=0)
#frame4.pack()

list_frame= tk.LabelFrame(frame4, text="BOOK LIST", padx=30, pady=20,
bg='#CAA74E', font=("Arial Black", ),)
list_frame.pack()

#scrollbar
my_scrollbar=ttk.Scrollbar(list_frame, orient='vertical')


#book list box

book_list=tk.Listbox(list_frame,  width=50, height=21 ,
selectmode=tk.EXTENDED, bg='#FFE9E1', fg='black', font=('times new roman
',  ))#613E30
book_list.pack(side='left'  )
book_list.configure(yscrollcommand = my_scrollbar.set)

#for book_list in book_entry :
#
 #   book_list.insert(0, values)
  #  book_list.insert(tk.END, f"{book_entry}: {date}")

bolded = font.Font( font=('times new roman', 9), weight='bold',) # will use
the default font
book_list.config(font=bolded)



my_scrollbar.configure(command=book_list.yview)
my_scrollbar.pack(side='right', fill='y')

update_book_list()

# book loan frame

frame5=tk.Frame(frame_container, bg='#907419', padx=68)


#canvas.create_window((0,0), window=frame5, anchor='nw')
frame5.grid(row=3,column=0, pady=30, )
#frame5.pack()

loan_frame= tk.LabelFrame(frame5, text="BOOK LOAN",bg='#907419', font=('arial
black', ) )
loan_frame.pack(side='left')
```

```
frame6=tk.Frame(frame5, bg='#907419')
frame6.pack(side='right')
info=tk.Label(frame6, text=' *Loan Info*', bg='#907419', font=('times new roman', ))
info_box=tk.Text(frame6, width=22, height=15, )
info.grid(row=0,column=0)
info_box.grid(row=1, column=0)
info_box.insert(1.0,"\n\n*Insert Info For loan\n\nOne Book=RM0.50 \n\nMax Loan=5\n\n*Book Must Be Returned After 3 Week\n\n*If book was loss the borrower need to pay  full price of the book")
info_box.configure(state='disabled')

name=tk.Label(loan_frame, text='Name', bg='#907419',  font=( 'Bahnschrift', 12))
name_entry=tk.Entry(loan_frame, bg='white')
name.grid(row=0, column=1)
name_entry.grid(row=1, column=1)


count=tk.Label(loan_frame, text='Book Count', bg='#907419',  font=( 'Bahnschrift', 12))
count_spinbox=tk.Spinbox(loan_frame, from_=0, to=5, bg='white')
count.grid(row=2, column=1)
count_spinbox.grid(row=3, column=1)

book=tk.Label(loan_frame, text='Book Title', bg='#907419', font=( 'Bahnschrift', 12))
book_entry=ttk.Combobox(loan_frame, values=[], )
book_entry.bind('<<ComboboxSelected>>', comboclick)
book.grid(row=4, column=1)
book_entry.grid(row=5, column=1)
#book_entry['values']= book_list.get
book_entry['values'] = tuple(book_list.get(0, tk.END))

book_loan=tk.Text(loan_frame, height=5, width=30, bg='white')
book_loan.grid(row=6, column=1, )

loan_button=tk.Button(loan_frame, text='LOAN', bg='#A28F65',  font=( 'Bahnschrift', 12), command= loan)
loan_button.grid(row=7, column=1, sticky='ew',  )

loan_entry=tk.Text(loan_frame, height=6, width=35, bg='#A28F65')
loan_entry.grid(row=8, column=1)


for widget in loan_frame.winfo_children():
    widget.grid_configure(padx=10, pady=5)
    loan_button.grid_configure(pady=25)
```

```



root.mainloop ()
```

4.0 GRAPHIC USER INTERFACE



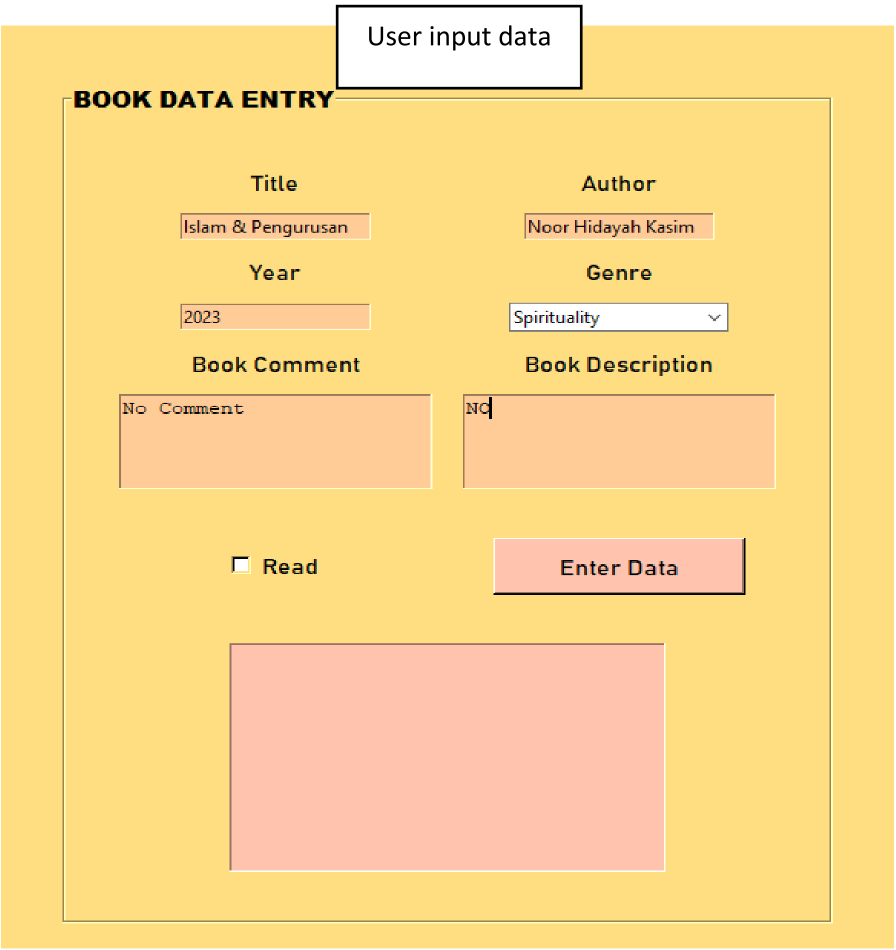
root title

Refresh root button

Library Organizer

Click the below button to refresh the window.

MY PERSONAL LIBRARY ORGANIZER



User input data

BOOK DATA ENTRY

Title: Islam & Pengurusan

Author: Noor Hidayah Kasim

Year: 2023

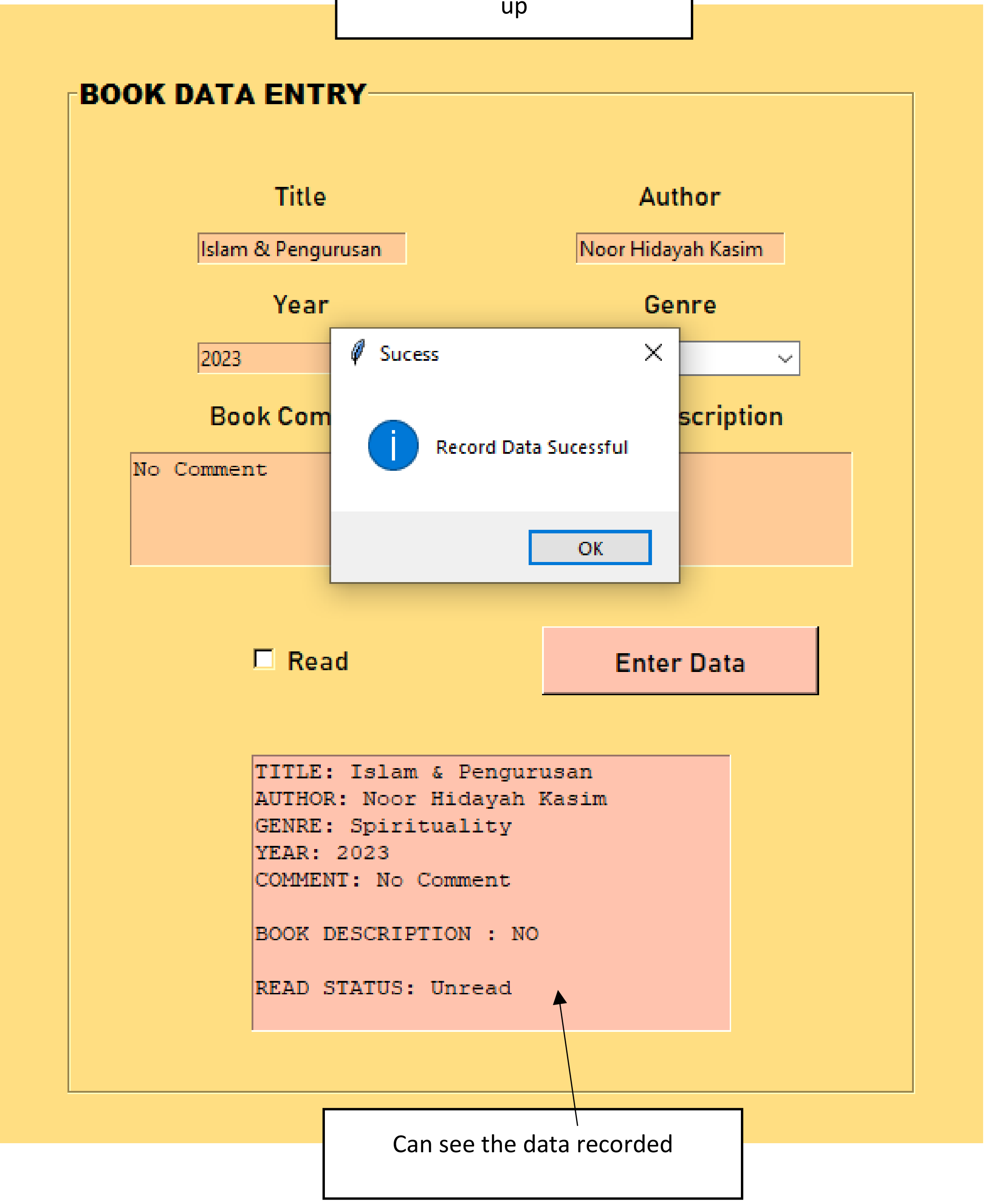
Genre: Spirituality

Book Comment: No Comment

Book Description: NO

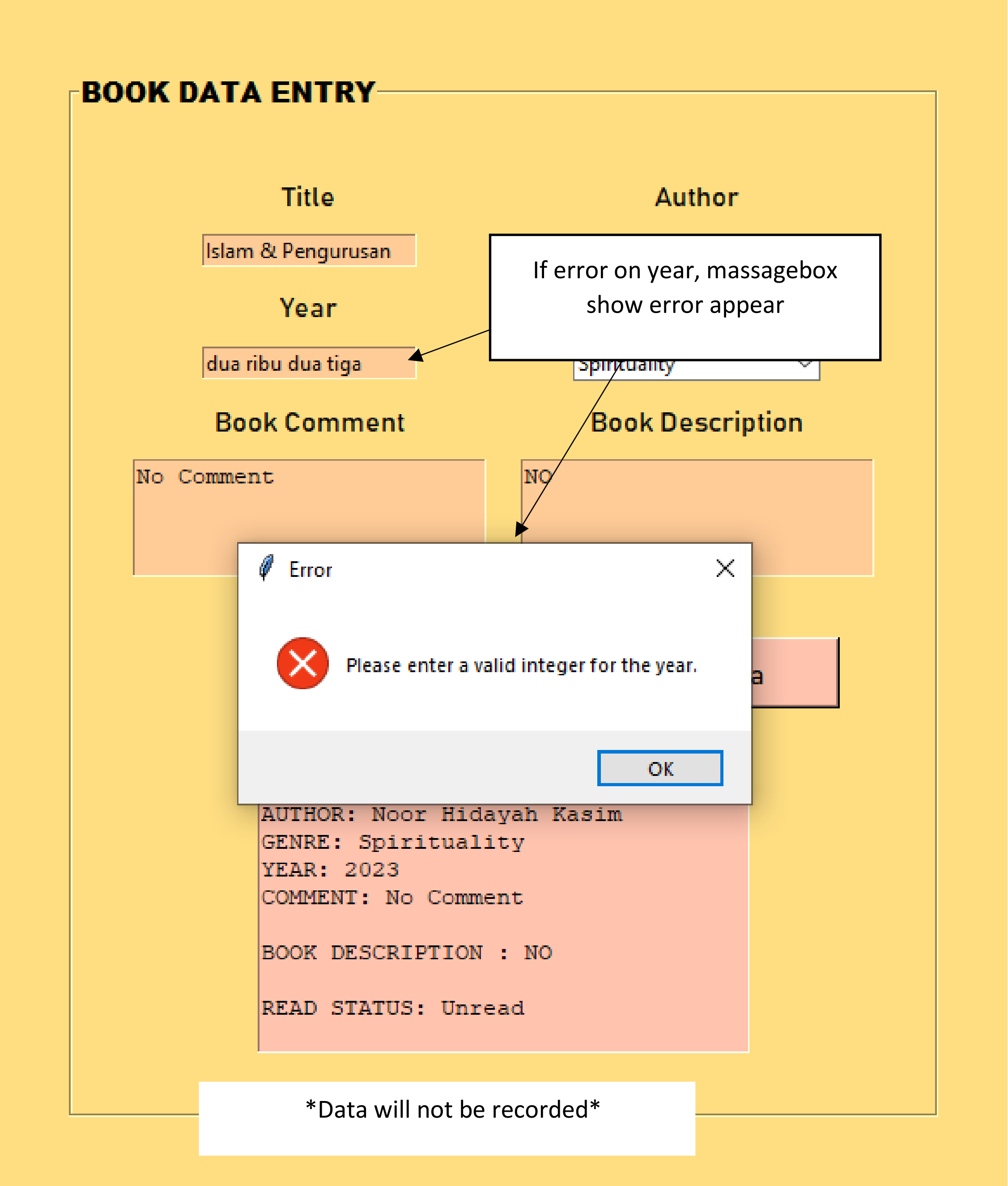
□ Read

If record data success massagebox show info pop up



```
TITLE: Islam & Pengurusan
AUTHOR: Noor Hidayah Kasim
GENRE: Spirituality
YEAR: 2023
COMMENT: No Comment

BOOK DESCRIPTION : NO

READ STATUS: Unread
```

Can see the data recorded



BOOK DATA ENTRY

Title: Islam & Pengurusan

Year: dua ribu dua tiga

Spirituality

Book Comment: No Comment

Book Description: NO

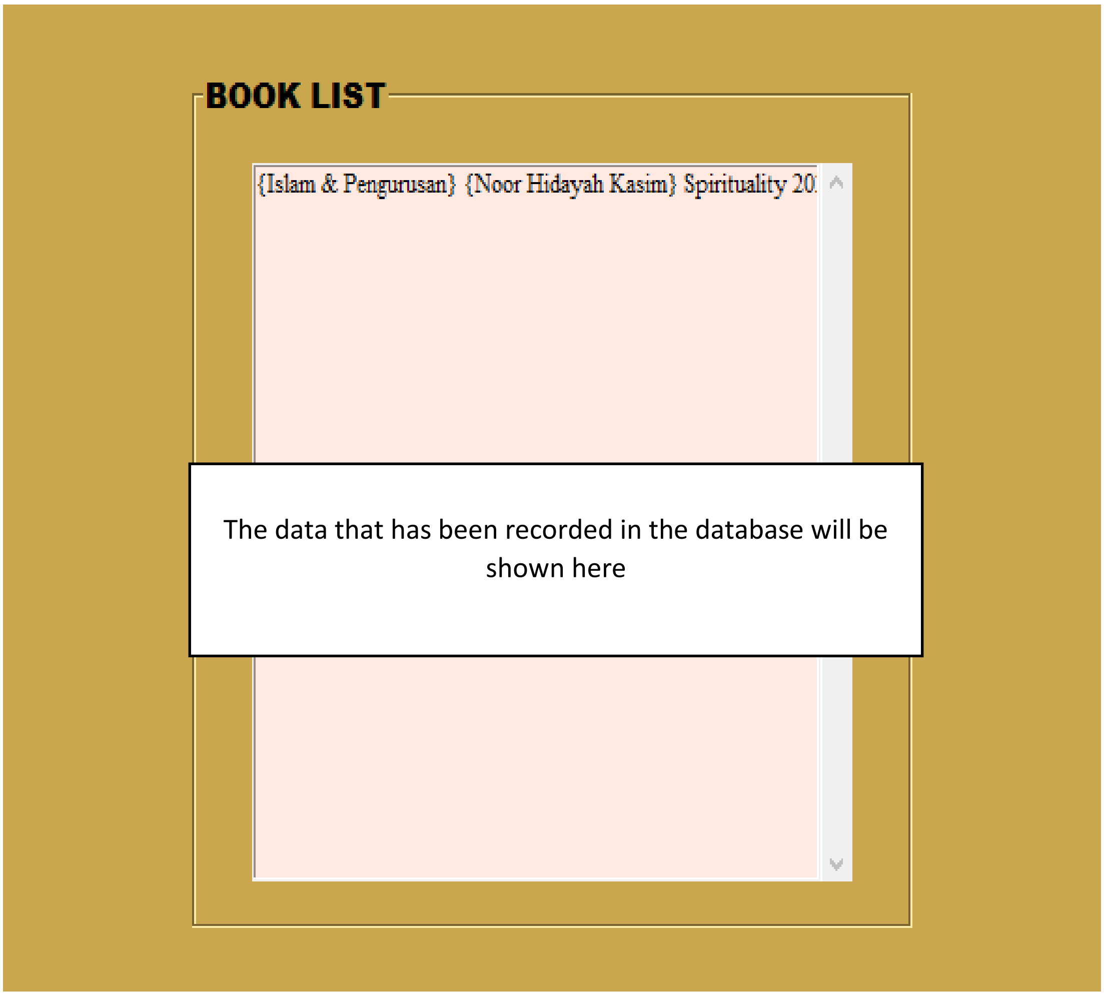
If error on year, massagebox show error appear

Error

Please enter a valid integer for the year.

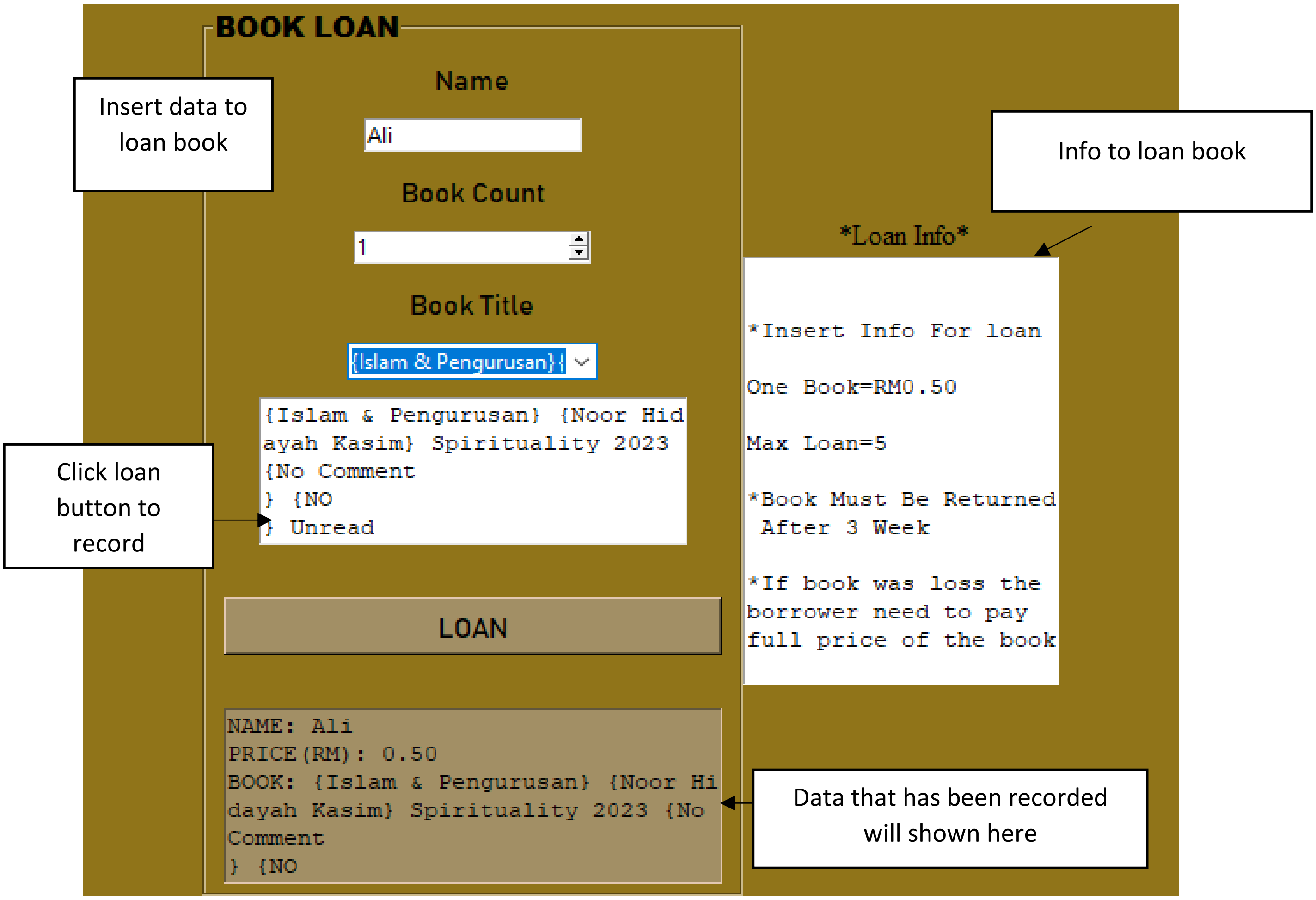
```
AUTHOR: Noor Hidayah Kasim
GENRE: Spirituality
YEAR: 2023
COMMENT: No Comment

BOOK DESCRIPTION : NO

READ STATUS: Unread
```

Data will not be recorded



{Islam & Pengurusan} {Noor Hidayah Kasim} Spirituality 20

The data that has been recorded in the database will be shown here



Name: Ali

Book Count: 1

Book Title: {Islam & Pengurusan}{

```
{Islam & Pengurusan} {Noor Hid
ayah Kasim} Spirituality 2023
{No Comment
} {NO
} Unread
```

Loan Info

```
*Insert Info For loan

One Book=RM0.50

Max Loan=5

*Book Must Be Returned
 After 3 Week

*If book was loss the
borrower need to pay
full price of the book
```

```
NAME: Ali
PRICE(RM): 0.50
BOOK: {Islam & Pengurusan} {Noor Hi
dayah Kasim} Spirituality 2023 {No
Comment
} {NO
```

Insert data to loan book

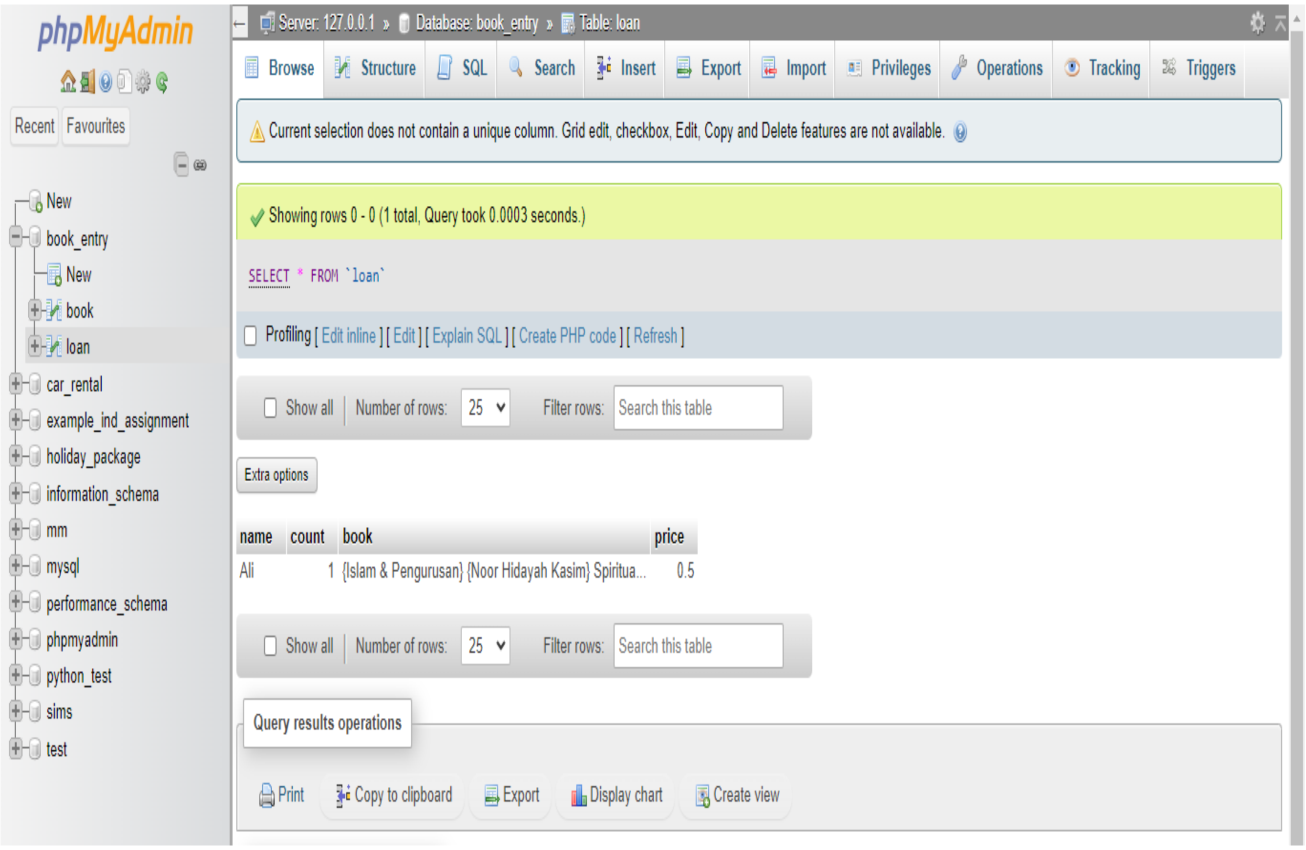
Click loan button to record

Info to loan book

Data that has been recorded will shown here

5.0 DATABASE

5.1 Browse



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

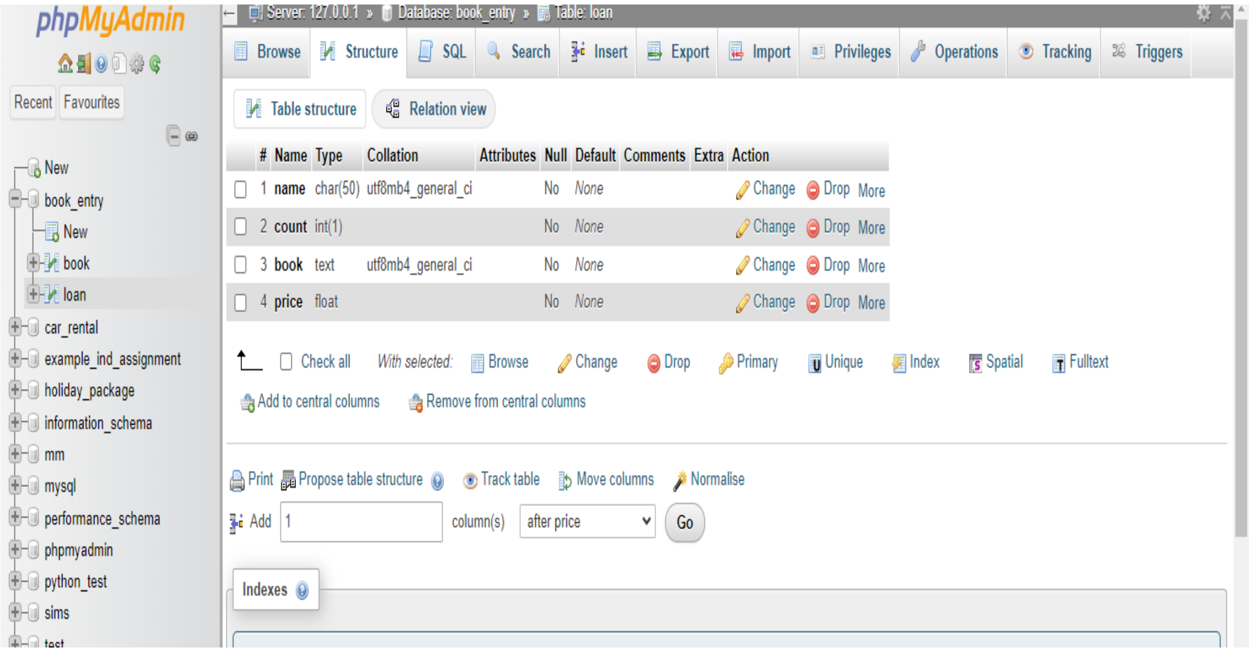
Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)

```
SELECT * FROM `loan`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all
Number of rows: 25
Filter rows: Search this table
Extra options

name	count	book	price
Ali	1	{Islam & Pengurusan} {Noor Hidayah Kasim} Spiritua...	0.5

5.2 Structure



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	name	char(50)	utf8mb4_general_ci		No	*None*			Change Drop More
□	2	count	int(1)			No	*None*			Change Drop More
□	3	book	text	utf8mb4_general_ci		No	*None*			Change Drop More
□	4	price	float			No	*None*			Change Drop More

6.0 CONCLUSION

In conclusion, developing the personal library organizer using Python and Tkinter has provided valuable insights into GUI programming and database integration. Through this project, I've gained practical experience in creating an interactive and user-friendly application for organizing and managing personal book collections. I hope to continue refining my skills in software development, exploring more advanced features, and contributing to larger projects in the future. This project has not only enhanced my understanding of Python and Tkinter but also sparked enthusiasm for further learning and application in the field of software development.

7.0 REFERENCE

Python Tkinter Tutorial. (2023, March 13). GeeksforGeeks. https://www.geeksforgeeks.org/python-tkinter-tutorial/

Python, R. (2023, January 30). *Python GUI Programming With Tkinter*. https://realpython.com/python-gui-tkinter/

Tkinter Tutorial. (2023, November 23). Python Tutorial - Master Python Programming for Beginners From Scratch. https://www.pythontutorial.net/tkinter/

Tkinter (GUI Programming) - Python Tutorial. (n.d.). https://pythonbasics.org/tkinter/

Binding Dropdown Menus and Combo Boxes - Python Tkinter GUI Tutorial #45. (2020, March 3). [Video]. YouTube. Retrieved January 4, 2024, from https://www.youtube.com/watch?v=OPUSBBD2OJw&t=3s