

```

import tkinter
from tkinter import *
from tkinter import ttk
import webbrowser
from tkcalendar import *
from time import *
from tkinter import messagebox

window = Tk()
window.rowconfigure(0, weight=1)
window.columnconfigure(0, weight=1)
window.title("Car Rental System")
window.state('zoomed')

page1 = Frame(window)
page2 = Frame(window)
page3 = Frame(window)
page4 = Frame(window)
page5 = Frame(window)
page6 = Frame(window)

for frame in (page1, page2, page3, page4, page5, page6):
    frame.grid(row=0, column=0, sticky='nsew')

def show_page(frame):
    frame.tkraise()

show_page(page1)

#***** Define
def exit_program():
    tkinter.messagebox.showinfo(title="Confirm?", message="Do you want to exit?")
    window.destroy()

def get_manual():
    printcar = f"Car Type: Manual Car"
    cartype.config(text=printcar)
    show_page(page4)

def get_auto():
    printcar = f"Car Type: Automatic Car"
    cartype.config(text=printcar)
    show_page(page3)

def cust_payment_manual():
    hour = hour_book_input2.get()
    day = day_input2.get()
    if (hour and day):
        hour = float(hour_book_input2.get())
        day = float(day_input2.get())
        car = v2.get().upper()
        price_dict = {
            'VIVA': [124.0, 6.0],
            'MYVI': [124.0, 6.0],
            'KANCIL': [100.0, 5.0]
        }

        if car in price_dict.keys():
            total_day2 = day * price_dict[car][0]
            total_hour2 = hour * price_dict[car][1]
            totalpayment = total_hour2 + total_day2

            printday = f"Payment for each Day: RM {total_day2}"
            payment_day.config(text=printday)

            printhour = f"Payment for each Hour: RM {total_hour2}"
            payment_hour.config(text=printhour)

```

```

        printtotal = f"Your Total Payment: RM {totalpayment}"
        payment_total.config(text=printtotal)
        print(totalpayment)
    else:
        totalpayment = 0
        print('failed')
        show_page(page5)
    else:
        tkinter.messagebox.showwarning(title="Error", message="Please fill all the
blank!")

def cust_payment_auto():
    hour = hour_book_input.get()
    day = day_input.get()
    if (hour and day):
        hour = float(hour_book_input.get())
        day = float(day_input.get())

        car = v.get().upper()
        price_dict = {
            'VIVA': [148.0, 7.0],
            'BEZZA': [190.0, 9.0],
            'ALZA': [240.0, 11.0]
        }

        if car in price_dict.keys():
            total_day = day * price_dict[car][0]
            total_hour = hour * price_dict[car][1]
            totalpayment = total_hour + total_day

            printday = f"Payment for each Day: RM {total_day}"
            payment_day.config(text=printday)

            printhour = f"Payment for each Hour: RM {total_hour}"
            payment_hour.config(text=printhour)

            printtotal = f"Your Total Payment: RM {totalpayment}"
            payment_total.config(text=printtotal)
            print(totalpayment)
        else:
            totalpayment = 0
            print('failed')
            show_page(page5)
    else:
        tkinter.messagebox.showwarning(title="Error", message="Please fill all the
blank!")

def grab_date_time():
    hour = hour_input.get()
    minute = minute_input.get()
    ampm = ampm_input.get()

    if hour and minute and ampm:

        ampm = ampm_input.get()
        minute = minute_input.get()
        hour = hour_input.get()

        printhour = f"{hour}:{minute} {ampm}"
        hour_receipt.config(text=printhour)

        date = cal.get_date()
        printdate = f"{date}"
        date_receipt.config(text=printdate)

        tkinter.messagebox.showinfo(title="Successfull", message="Your Booking Date and

```

```

Time Saved!")
    else:
        tkinter.messagebox.showwarning(title="Error", message="You need fill all the
blank!")

def update_time():
    time = strftime("%I:%M:%S %p")
    clocks.config(text=time)

    clocks.after(1000,update_time)

def choose():

    if (v.get() != ''):
        carmodel = v.get()
        print_carmodel = f"Car Model: {carmodel}"
        receipt_carmodel.config(text=print_carmodel)
    elif (v2.get() != ''):
        carmodel = v2.get()
        print_carmodel = f"Car Model: {carmodel}"
        receipt_carmodel.config(text=print_carmodel)
    else:
        print("Mission failed")

v=StringVar()
v2=StringVar()

def user_info():
    #user info
    name1 = name_string.get()
    gender = gender_input.get()
    age = age_input.get()
    phone = phone_input.get()
    term = term_var.get()

    if term == "Agree":
        if name1 and gender and age and phone:
            name1 = name_string.get()
            printname = f"Name: {name1}"
            receipt_name_label.config(text=printname)
            gender = gender_input.get()
            printgender = f"Gender: {gender}"
            receipt_gender_label.config(text=printgender)

            age = age_input.get()
            printage = f"Age: {age}"
            receipt_age_label.config(text=printage)

            phone = phone_input.get()
            printphone = f"Phone Num: {phone}"
            receipt_phone_label.config(text=printphone)

            term = term_var.get()
            printterm = f"You {term} with our term and condition"
            receipt_term_label.config(text=printterm)
            tkinter.messagebox.showinfo(title="Successfull", message="Your Data
Saved!")
            show_page(page2)
        else:
            tkinter.messagebox.showwarning(title="Error", message="You need to fill all
the blank")
    else:
        tkinter.messagebox.showwarning(title="Error", message="You need to accept the
term")

```

```

# Display Welcome
photo = PhotoImage(file='car_logo-removebg-preview.png')
page1_Label = Label(page1, text="Welcome To Car Rental System", font=('Sans
open',20,'bold'), bg="#a29bfe", fg='white', relief=RIDGE, bd=5, padx=20, pady=10,
image=photo, compound='left')
page1_Label.place(x=440, y=10)

# save user input
user_info_frame = LabelFrame(page1, text= "User Information", font=('Sans
open',10,'bold'), bg="#6c5ce7", fg="white", relief=RIDGE, bd=5, padx=88, pady=10)
user_info_frame.place(x=440, y=268)

name_label = tkinter.Label(user_info_frame, text="Name", font=('Sans open',10,'bold'),
bg="#6c5ce7", fg="white")
name_label.grid(column=0, row=0)
name_string = tkinter.StringVar()
name_user = tkinter.Entry(user_info_frame, textvariable=name_string)
name_user.grid(column=0, row=1)

gender_label = tkinter.Label(user_info_frame, text="Gender", font=('Sans
open',10,'bold'), bg="#6c5ce7", fg="white")
gender_input = ttk.Combobox(user_info_frame, values=["Male","Female"])
gender_label.grid(column=1, row=0)
gender_input.grid(column=1, row=1)

age_label = tkinter.Label(user_info_frame, text="Age", font=('Sans open',10,'bold'),
bg="#6c5ce7", fg="white")
age_input = tkinter.Spinbox(user_info_frame, from_=17, to=85)
age_label.grid(column=2, row=0)
age_input.grid(column=2, row=1)

phone_label = tkinter.Label(user_info_frame, text="Phone Number", font=('Sans
open',10,'bold'), bg="#6c5ce7", fg="white")
phone_label.grid(column=0, row=3)
phone_input = tkinter.Entry(user_info_frame)
phone_input.grid(column=0, row=4)

for widget in user_info_frame.winfo_children():
    widget.grid_configure(padx=20, pady=5)

#accept term
term_frame = tkinter.LabelFrame(page1, text= "Terms & Condition", bg="#6c5ce7",
fg="White", relief=RIDGE, bd=5, padx=250, pady=5)
term_frame.place(x=440, y=435)

term_var = tkinter.StringVar(value="Don't Agree")
term_check = tkinter.Checkbutton(term_frame, text= "I accept the term and condition.",
                                variable= term_var, onvalue= "Agree", offvalue= "Don't
Agree",
                                bg="#6c5ce7", activebackground="#6c5ce7")
term_check.grid(column=0, row=0)

#button
button_info = tkinter.Button(page1, text= "Submit Data", command= user_info,
bg="#a29bfe", relief=RIDGE, bd=5, padx=313, pady=0)
button_info.place(x=440, y=493)

#***** Page 2
#header page 2
page2_Label = Label(page2, text="Car Booking", font=('Sans open',20,'bold'),
bg="#a29bfe", fg='white', relief=RIDGE, bd=5, padx=103, pady=10, image=photo,
compound='left')
page2_Label.place(x=440, y=10)

# type of car user need
car_frame = LabelFrame(page2, text= "Car Type (Auto/Manual)", bg="#a29bfe",
relief=RIDGE, bd=5, padx=228, pady=10)

```

```

car_frame.place(x=440, y=542)

type_input = tkinter.Button(car_frame, text= "Automatic Car", bg="#6c5ce7",
activebackground="#6c5ce7", command=get_auto)
type_input.grid(column=0,row=1)

type2_input = tkinter.Button(car_frame, text= "Manual Car", bg="#6c5ce7",
activebackground="#6c5ce7", command=get_manual)
type2_input.grid(column=1,row=1)

for widget in car_frame.winfo_children():
    widget.grid_configure(padx=20,pady=5)

# calander
cal_frame = LabelFrame(page2, text="Booking Date & Time", bg="#a29bfe", relief=RIDGE,
bd=5, padx=22, pady=5)
cal_frame.place(x=440, y=268)

cal = Calendar(cal_frame, selectmode="day", year=2023, month=1, day=24)
cal.grid(column=0,row=0)

clocks = Label(cal_frame, font=("Open sans",20), fg="Black", bg="#a29bfe")
clocks.grid(column=0,row=1)
update_time()

for widget in cal_frame.winfo_children():
    widget.grid_configure(padx=20,pady=5)

# time
time_frame = LabelFrame(cal_frame, bg="#6c5ce7", relief=RIDGE, bd=5, padx=0, pady=23)
time_frame.grid(column=1,row=0)

hour_label = Label(time_frame, text="Time To Book (Hour)", bg="#6c5ce7")
hour_label.grid(column=0,row=0)
hour_input = tkinter.Entry(time_frame)
hour_input.grid(column=0,row=1)

minute_label = Label(time_frame, text="Time To Book (Minute)", bg="#6c5ce7")
minute_label.grid(column=0,row=2)
minute_input = tkinter.Entry(time_frame)
minute_input.grid(column=0,row=3)

ampm_label = Label(time_frame, bg="#6c5ce7")
ampm_label.grid(column=1,row=0)
ampm_input = ttk.Combobox(time_frame, values=["AM","PM"])
ampm_input.grid(column=1,row=1)

cal_button = Button(time_frame, text="Book Date & Time", bg="#6c5ce7",
activebackground="#6c5ce7", command=grab_date_time)
cal_button.grid(column=1,row=3)

for widget in time_frame.winfo_children():
    widget.grid_configure(padx=20,pady=5)

#***** Page 3
#header page 3
page3_Label = Label(page3,text="Automatic Car", font=('Sans open',20,'bold'),
bg="#a29bfe",fg='white', relief=RIDGE, bd=5, padx=96, pady=10, image=photo,
compound='left')
page3_Label.place(x=440, y=10)

auto_frame = LabelFrame(page3, text= "Choose Car Model", bg="#a29bfe", relief=RIDGE,
bd=5, padx=47, pady=10)
auto_frame.place(x=440, y=268)

#viva section
photo_viva = PhotoImage(file='rsz_2perodua-viva-front.png')

```

```

viva_label = tkinter.Label(auto_frame, text="Perodua Viva", bg="#a29bfe")
viva_label.grid(column=0, row=0)
viva_input = tkinter.Radiobutton(auto_frame, image=photo_viva, variable=v,
value='Viva', tristatevalue=0, bg="#a29bfe", activebackground="#a29bfe",
command=choose)
viva_input.grid(column=0, row=1)

#bezza section
photo_bezza = PhotoImage(file='rsz_1rsz_1bezza-removebg-preview.png')
bezza_label = tkinter.Label(auto_frame, text="Perodua Bezza", bg="#a29bfe")
bezza_label.grid(column=1, row=0)
bezza_input = tkinter.Radiobutton(auto_frame, image=photo_bezza, variable=v,
value='Bezza', tristatevalue=0, bg="#a29bfe", activebackground="#a29bfe",
command=choose)
bezza_input.grid(column=1, row=1)

#alza section
photo_alza = PhotoImage(file='rsz_1rsz_1alza__1_-removebg-preview.png')
alza_label = tkinter.Label(auto_frame, text="Perodua Alza", bg="#a29bfe")
alza_label.grid(column=2, row=0)
alza_input = tkinter.Radiobutton(auto_frame, image=photo_alza, variable=v,
value='Alza', tristatevalue=0, bg="#a29bfe", activebackground="#a29bfe",
command=choose)
alza_input.grid(column=2, row=1)

# choose day or hour
book_frame = LabelFrame(page3, text="Booking duration", bg="#a29bfe", relief=RIDGE,
bd=5, padx=90, pady=4)
book_frame.place(x=440, y=459)

day_label = Label(book_frame, text="How Many Day", bg="#a29bfe")
day_label.grid(column=0, row=0)
day_input = Entry(book_frame)
day_input.grid(column=0, row=1)

hour_book_label = Label(book_frame, text="How Many Hour", bg="#a29bfe")
hour_book_label.grid(column=0, row=2)
hour_book_input = Entry(book_frame)
hour_book_input.grid(column=0, row=3)

for widget in book_frame.winfo_children():
    widget.grid_configure(padx=20, pady=11)

#info price
price_frame = LabelFrame(page3, text="Price Information", bg="#6c5ce7", relief=RIDGE,
bd=5, padx=19, pady=0)
price_frame.place(x=796, y=459)

price_label = Label(price_frame, text="Perodua Viva : RM7/Hour - RM148/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label.grid(column=0, row=0, sticky='w')
price_label1 = Label(price_frame, text="Perodua Bezza : RM9/Hour - RM190/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label1.grid(column=0, row=1, sticky='w')
price_label2 = Label(price_frame, text="Perodua Alza : RM11/Hour - RM240/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label2.grid(column=0, row=2, sticky='w')

calc_payment = Button(price_frame, text="Calculate Payment", command=cust_payment_auto,
bg="#a29bfe", activebackground="#a29bfe", relief=RIDGE, bd=5, padx=20, pady=0)
calc_payment.grid(column=0, row=3)

for widget in price_frame.winfo_children():
    widget.grid_configure(padx=15, pady=10)

#***** Page 4
# header page 4

```

```

page3_Label = Label(page4, text="Manual Car", font=('Sans open', 20, 'bold'),
bg="#a29bfe", fg='white', relief=RIDGE, bd=5, padx=109, pady=10, image=photo,
compound='left')
page3_Label.place(x=440, y=10)

manual_frame = LabelFrame(page4, text= "Choose Car Model", bg="#a29bfe", relief=RIDGE,
bd=5, padx=75, pady=13)
manual_frame.place(x=440, y=268)

#viva section
photo_viva2 = PhotoImage(file='rsz_2perodua-viva-front.png')
viva_label2 = tkinter.Label(manual_frame, text="Perodua Viva", bg="#a29bfe")
viva_label2.grid(column=0, row=0)
viva_input2 = tkinter.Radiobutton(manual_frame, image=photo_viva2, variable=v2,
value='Viva', tristatevalue=0, bg="#a29bfe", command=choose)
viva_input2.grid(column=0, row=1)

# myvi section
photo_myvi2 = PhotoImage(file='rsz_1myvi.png')
myvi_label = tkinter.Label(manual_frame, text="Perodua Myvi", bg="#a29bfe")
myvi_label.grid(column=1, row=0)
myvi_input = tkinter.Radiobutton(manual_frame, image=photo_myvi2, variable=v2,
value='Myvi', tristatevalue=0, bg="#a29bfe", command=choose)
myvi_input.grid(column=1, row=1)

# kancil section
photo_kancil = PhotoImage(file='rsz_1rsz_kancil.png')
kancil_label = tkinter.Label(manual_frame, text="Perodua Kancil", bg="#a29bfe")
kancil_label.grid(column=2, row=0)
kancil_input = tkinter.Radiobutton(manual_frame, image=photo_kancil, variable=v2,
value='Kancil', tristatevalue=0, bg="#a29bfe", command=choose)
kancil_input.grid(column=2, row=1)

# choose day or hour
book_frame2 = LabelFrame(page4, text="Booking duration", bg="#a29bfe", relief=RIDGE,
bd=5, padx=90, pady=4)
book_frame2.place(x=440, y=459)

day_label2 = Label(book_frame2, text="How Many Day", bg="#a29bfe")
day_label2.grid(column=0, row=0)
day_input2 = Entry(book_frame2)
day_input2.grid(column=0, row=1)

hour_book_label2 = Label(book_frame2, text="How Many Hour", bg="#a29bfe")
hour_book_label2.grid(column=0, row=2)
hour_book_input2 = Entry(book_frame2)
hour_book_input2.grid(column=0, row=3)

for widget in book_frame2.winfo_children():
    widget.grid_configure(padx=20, pady=11)

#info price manual
price_frame2 = LabelFrame(page4, text= "Price Information", bg="#6c5ce7", relief=RIDGE,
bd=5, padx=23, pady=0)
price_frame2.place(x=796, y=459)

price_label = Label(price_frame2, text="Perodua Viva : RM6/Hour - RM124/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label.grid(column=0, row=0, sticky='w')
price_label1 = Label(price_frame2, text="Perodua Myvi : RM6/Hour - RM124/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label1.grid(column=0, row=1, sticky='w')
price_label2 = Label(price_frame2, text="Perodua Kancil : RM5/Hour - RM100/Day",
font=('Sans Open', 10, 'bold'), bg="#6c5ce7", fg="white", justify=LEFT, anchor='w')
price_label2.grid(column=0, row=2, sticky='w')

```

```

go_to_payment2 = Button(price_frame2, text="Calculate Payment", command=
cust_payment_manual, bg="#a29bfe", activebackground="#a29bfe", relief=RIDGE, bd=5,
padx=20, pady=0 )
go_to_payment2.grid(column=0, row=3)

for widget in price_frame2.winfo_children():
    widget.grid_configure(padx=15, pady=10)

#***** Page 5

page5_Label = Label(page5, text="Receipt", font=('Sans open', 20, 'bold'), bg="#a29bfe",
fg="white", relief=RIDGE, bd=5, padx=125, pady=10, image=photo, compound='left')
page5_Label.place(x=440, y=10)

#main frame
empty_frame = LabelFrame(page5, bg="#a29bfe", relief=RIDGE, bd=5, padx=10, pady=10)
empty_frame.place(x=440, y=267)

#user info
receipt_frame = LabelFrame(empty_frame, text="Customer Information", bg="#a29bfe",
relief=RIDGE, padx=90, pady=10)
receipt_frame.grid(column=0, row=0)

receipt_name_label = Label(receipt_frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_name_label.grid(column=0, row=0, sticky='w')

receipt_gender_label = Label(receipt_frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_gender_label.grid(column=0, row=1, sticky='w')

receipt_age_label = Label(receipt_frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_age_label.grid(column=1, row=0, sticky='w')

receipt_phone_label = Label(receipt_frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_phone_label.grid(column=1, row=1, sticky='w')

receipt_term_label = Label(receipt_frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_term_label.grid(column=0, row=2, sticky='w')

for widget in receipt_frame.winfo_children():
    widget.grid_configure(padx=15, pady=10)

# booking time & date resit
receipt_frame2 = LabelFrame(empty_frame, text="Your Booking Date & Time", bg="#a29bfe",
relief=RIDGE, padx=100, pady=10)
receipt_frame2.grid(column=0, row=1)

hour_receipt = Label(receipt_frame2, bg="#a29bfe")
hour_receipt.grid(column=1, row=0)

date_receipt = Label(receipt_frame2, bg="#a29bfe")
date_receipt.grid(column=0, row=0)

# car type
payment_Frame = LabelFrame(empty_frame, text="Your Payment", bg="#a29bfe",
relief=RIDGE, padx=100, pady=10)
payment_Frame.grid(column=0, row=2)

cartype = Label(payment_Frame, bg="#a29bfe", justify=LEFT, anchor="w")
cartype.grid(column=0, row=0, sticky='w')

receipt_carmodel = Label(payment_Frame, bg="#a29bfe", justify=LEFT, anchor="w")
receipt_carmodel.grid(column=0, row=1, sticky='w')

payment_day = Label(payment_Frame, bg="#a29bfe", justify=LEFT, anchor="w")
payment_day.grid(column=0, row=2, sticky='w')

payment_hour = Label(payment_Frame, bg="#a29bfe", justify=LEFT, anchor="w")

```



```

payment_hour.grid(column=0,row=3, sticky='w')

payment_total = Label(payment_Frame, bg="#a29bfe", justify=LEFT, anchor="w")
payment_total.grid(column=0,row=4, sticky='w')

pay_button = Button(empty_frame, text="Proceed to Payment", bg="#a29bfe",
activebackground="#a29bfe", command= lambda: show_page(page6))
pay_button.grid(column=0,row=3)

#***** Page 6

page6_Label = Label(page6,text="Payment", font=('Sans open',20,'bold'), bg="#a29bfe",
fg="white", relief=RIDGE, bd=5, padx=125, pady=10, image=photo, compound='left')
page6_Label.place(x=440, y=10)

link_Frame = LabelFrame(page6, text="Click to pay", bg="#a29bfe", relief=RIDGE, bd=5,
padx=242, pady=10 )
link_Frame.place(x=440, y=269)

cimb_photo = PhotoImage(file='rsz_cimb-clicks-logo-vector-removebg-preview.png')
bi_photo = PhotoImage(file='rsz_bank_islam-removebg-preview.png')

my_link2 = Label(link_Frame, image=bi_photo, bg="#a29bfe", fg="Red", cursor='hand2')
my_link2.grid(column=0,row=1)
my_link2.bind('<Button-1>', lambda x:webbrowser.open_new("https://www.bankislam.biz/"))

my_link = Label(link_Frame, image=cimb_photo, bg="#a29bfe", fg="Red", cursor='hand2')
my_link.grid(column=0,row=0)
my_link.bind('<Button-1>', lambda
x:webbrowser.open_new("https://www.cimbclicks.com.my/clicks/#/"))

exit_button = Button(link_Frame, text="Exit", bg="#a29bfe", activebackground="#a29bfe",
padx=20, command=exit_program)
exit_button.grid(column=0,row=2)

for widget in link_Frame.wininfo_children():
    widget.grid_configure(pady=10)

window.mainloop()

```