UNIVERSITI TEKNOLOGI MARA

KEDAH BRANCH

SCHOOL OF INFORMATION SCIENCE

COLLEGE OF COMPUTING, INFORMATICS AND MATHEMATHICS

DIPLOMA IN LIBRARY INFORMATICS (CDIM144)

IML 208 : PROGRAMMING FOR LIBRARIES

INDIVIDUAL PROJECT : CAB BOOKING SYSTEM

PREPARED BY :

NURUL ALIEYA NATASA BINTI ALIAS SANI

(2022487572)

GROUP KCDIM1443D

PREPARED FOR :

SIR MOHD FIRDAUS MOHD HELMI

SUBMISSION DATE :

December 2023

INDIVUDUAL PROJECT : CAB BOOKING SYSREM

PREPARED BY

NURUL ALIEYA NATASA BINTI ALIS SANI

GROUP KCDIM1443D

IM144 – DIPLOMA IN LIBRARY INFORMATICS

SCHOOL IF INFORMATION SCIENCE

COLLEGE OF COMPUTING, INFORMATICS AND MATHEMATICS

UNIVERSITI TEKNOLOGI MARA (UITM)

KEDAH BRANCH

PROJECT NAME : Cab Booking System

FILE NAME : bookingsystem.py

PROMPT DATA :

1. **Name :** name of customer
2. **Surename :** surename of customer
3. **Address :** address of customer
4. **Postcode :** postcode customer
5. **Telephone :** no. telephone customer for contact
6. **Email :** email for sent receipt
7. **Location :** location to pick up and drop

FUCTION :

1. **CREATE**
2. **READ**
3. **RESERT**
4. **EXIT**

CONDITIONAL STATEMENT :YES

If-else statement

```
def Kilo():
    if var2.get() == 0:
        self.txtKm.configure(state=DISABLED)
        Km.set("0")
    elif var2.get() == 1 and varl1.get() != "" and varl2.get() != "":
        self.txtKm.configure(state=NORMAL)
        if varl1.get() == "UiTM":
            switch ={"Amanjaya Mall": 12,"Central Square": 13,"KTM SP":15,"UiTM": 0}
            Km.set(switch[varl2.get()])
        elif varl1.get() == "Amanjaya Mall":
            switch ={"Amanjaya Mall": 0,"Central Square": 6,"KTM SP":5,"UiTM": 12}
            Km.set(switch[varl2.get()])
        elif varl1.get() == "Central Square":
            switch ={"Amanjaya Mall": 6,"Central Square": 0,"KTM SP":1,"UiTM": 13}
            Km.set(switch[varl2.get()])
        elif varl1.get() == "KTM SP":
            switch ={"Amanjaya Mall": 5,"Central Square": 1,"KTM SP":0,"UiTM": 15}
            Km.set(switch[varl2.get()])
```

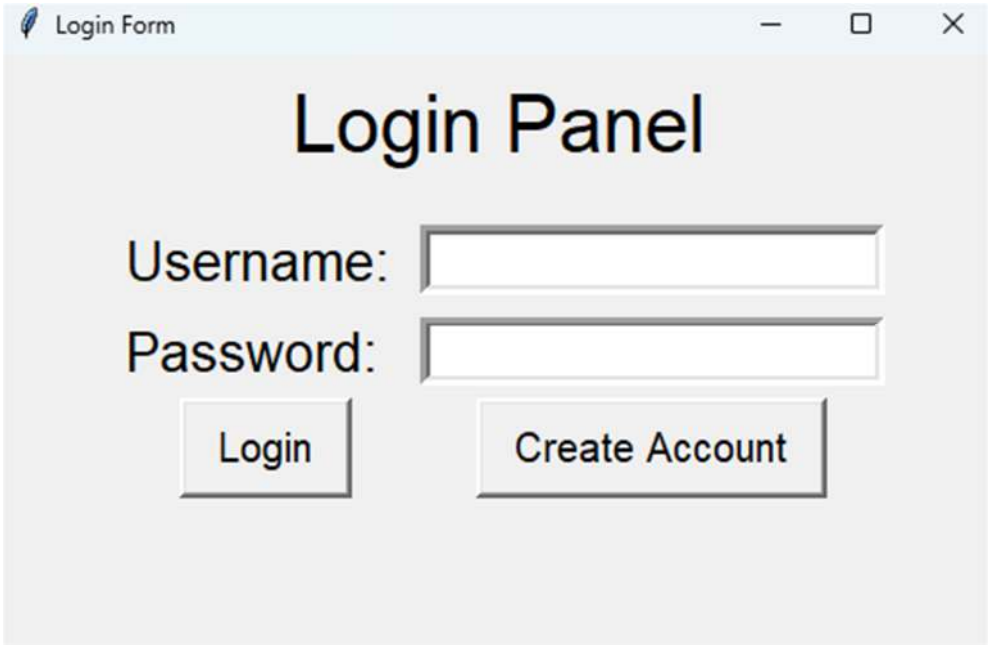
```
def new_user(self):
    #Establish Connection
    with sqlite3.connect('Users.db') as db:
        c = db.cursor()

    #Find Existing username if any take proper action
    find_user = ('SELECT * FROM user WHERE username = ?')
    c.execute(find_user,[(self.username.get())])
    if c.fetchall():
        ms.showerror('Error!','Username Already Taken!')
    else:
        ms.showinfo('Success!','Account Created!')
        self.log()
```

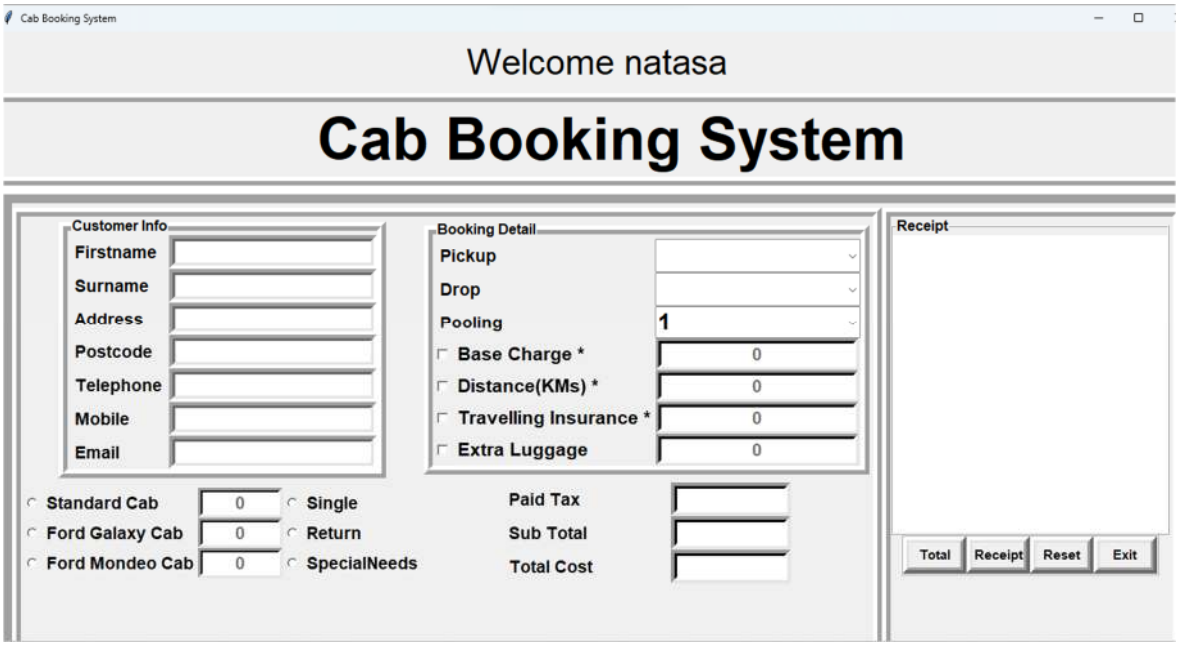
```
ngsystem.py
> User > OneDrive > Documents > SEM 3 UITM > IML208 > bookingsystem.py > travel > __init__ > iExit
            switch ={"Amanjaya Mall": 5,"Central Square": 1,"KTM SP":0,"UiTM": 15}
            Km.set(switch[var12.get()])


    def Travelling():
        global Item3
        if var3.get() == 1:
            self.txtTravel_Ins.configure(state = NORMAL)
            Item3=float(2)
            Travel_Ins.set("RM " + str(Item3))
        elif var3.get() == 0:
            self.txtTravel_Ins.configure(state = DISABLED)
            Travel_Ins.set("0")
            Item3=0


    def Lug():
        global Item4
        if (var4.get()==1):
            self.txtLuggage.configure(state = NORMAL)
            Item4=float(10)
            Luggage.set("RM "+ str(Item4))
        elif var4.get()== 0:
            self.txtLuggage.configure(state = DISABLED)
            Luggage.set("0")
            Item4=0
```

```
gsystem.py
> User > OneDrive > Documents > SEM 3 UITM > IML208 > bookingsystem.py > travel > __init__ > iExit
            Item4=0


    def selectCar():
        global Item5
        if carType.get() == 1:
            self.txtFordGalaxy.configure(state = DISABLED)
            FordGalaxy.set("0")
            self.txtFordMondeo.configure(state = DISABLED)
            FordMondeo.set("0")
            self.txtStandard.configure(state = NORMAL)
            Item5 = float(8)
            Standard.set("RM "+ str(Item5))
        elif carType.get() == 2:
            self.txtStandard.configure(state =DISABLED)
            Standard.set("0")
            self.txtFordMondeo.configure(state = DISABLED)
            FordMondeo.set("0")
            self.txtFordGalaxy.configure(state = NORMAL)
            Item5 = float(13)
            FordGalaxy.set("RM "+ str(Item5))
        else:
            self.txtStandard.configure(state =DISABLED)
            Standard.set("0")
            self.txtFordGalaxy.configure(state = DISABLED)
            FordGalaxy.set("0")
            self.txtFordMondeo.configure(state = NORMAL)
            Item5 = float(15)
            FordMondeo.set("RM "+ str(Item5))
```

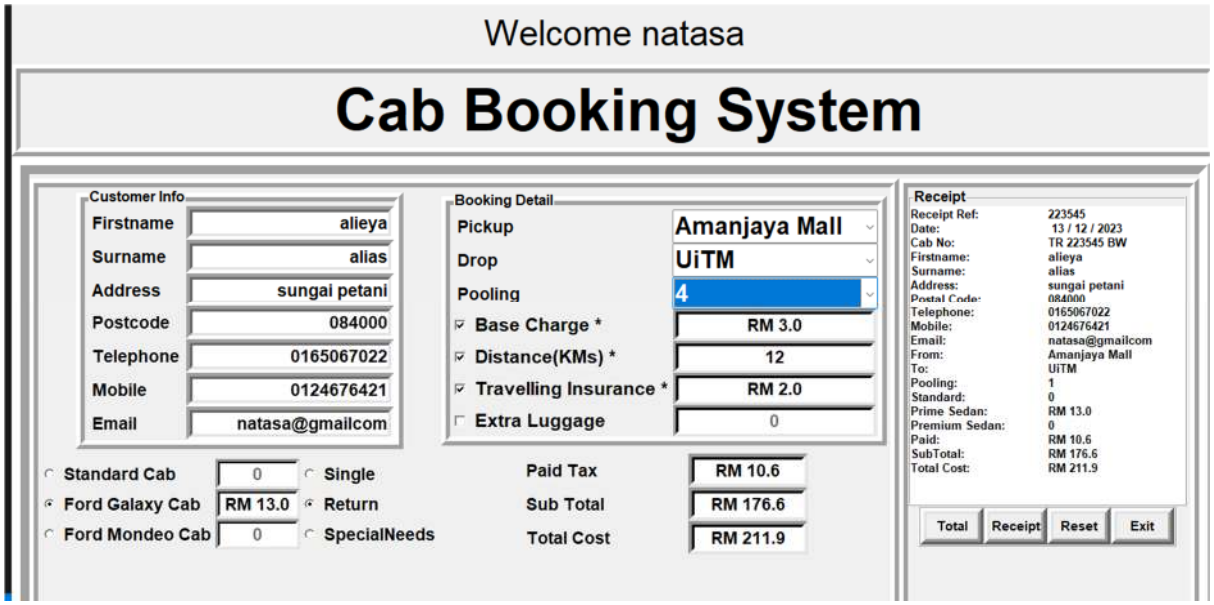
```
def Total_Paid():
    if ((var1.get() == 1 and var2.get() == 1 and var3.get() == 1 or var4.get() == 1) and carType.get() != 0 and journeyType.get() != 0 an
        if journeyType.get()==1:
            Item2=Km.get()
            Cost_of_fare = (Item1+(float(Item2)*Item5)+Item3+Item4)

            Tax = "RM " + str('%.1f'%((Cost_of_fare) *0.03))
            ST = "RM " + str('%.1f'%((Cost_of_fare)))
            TT = "RM " + str('%.1f'%(Cost_of_fare+((Cost_of_fare)*0.2)))
        elif journeyType.get()==2:
            Item2=Km.get()
            Cost_of_fare = (Item1+(float(Item2)*Item5)*1.1+Item3+Item4)

            Tax = "RM " + str('%.1f'%((Cost_of_fare) *0.06))
            ST = "RM " + str('%.1f'%((Cost_of_fare)))
            TT = "RM " + str('%.1f'%(Cost_of_fare+((Cost_of_fare)*0.2)))
        else:
            Item2=Km.get()
            Cost_of_fare = (Item1+(float(Item2)*Item5)*2+Item3+Item4)

            Tax = "RM " + str('%.1f'%((Cost_of_fare) *0.09))
            ST = "RM " + str('%.1f'%((Cost_of_fare)))
            TT = "RM " + str('%.1f'%(Cost_of_fare+((Cost_of_fare)*0.2)))

        PaidTax.set(Tax)
        SubTotal.set(ST)
        TotalCost.set(TT)
    else:
        w = ms.showwarning("Error !","Invalid Input\nPlease try again !!!")
```

GUI : YES


Login Form
Login Panel
Username:
Password:
Login
Create Account


Cab Booking System
Welcome natasa
Cab Booking System
Customer Info
Firstname
Surname
Address
Postcode
Telephone
Mobile
Email
Booking Detail
Pickup
Drop
Pooling
1
Base Charge *
Distance(KMs) *
Travelling Insurance *
Extra Luggage
Receipt
Standard Cab
Ford Galaxy Cab
Ford Mondeo Cab
Single
Return
SpecialNeeds
Paid Tax
Sub Total
Total Cost
Total
Receipt
Reset
Exit

RESULTS :



STRENGHT :

1. Code Organization:

- Consider organizing your code into multiple modules or classes to improve readability and maintainability.
- Use functions or methods to encapsulate logical blocks of code, making it easier to understand.

2. User Interface:

- Improve the user interface by providing clear instructions and labels for the input fields and buttons.
- Add comments in your code to explain the purpose and functionality of each section.

3. Database Handling:

- Consider adding error handling and validation when interacting with the database. For example, check if the connection to the database is successful and handle potential errors gracefully.

4. Security:

- Implement secure password handling. Storing passwords in plain text is not recommended. Consider using techniques like hashing for storing and validating passwords.

5. Error Handling:

- Implement comprehensive error handling to handle unexpected situations and provide meaningful error messages to the users.

6. Code Reusability:

- Try to write reusable code by creating functions or classes for common tasks.

7. Consistent Naming:

- Maintain a consistent naming convention for variables and functions. This enhances code readability.

8. User Feedback:

- Provide informative messages to the users, especially when errors occur. This helps users understand what went wrong and how to fix it.

9. Testing:

- Test your code thoroughly with different scenarios to ensure its robustness. Consider using unit tests to automate the testing process
  .

10. Documentation:

- Add comments and docstrings to explain the purpose of your functions, classes, and modules.

KAIZEN (ROOM FOR IMPROVEMENT) :

1. Need to imrpove code for backgroung color
2. Need to specific the place
3. Need more place for student