School of Information Science,

College of Computing, Informatics, and Mathematics,

Universiti Teknologi mara,

Merbok, Kedah.

Diploma in Library Informatics (CDIM144)

Programming for Library (IML208)

"Report in Python Program: Cat Owner Registration"

Prepared by:

Nurul Sofiyyah Binti Azhar (2022810368)

Class: KCDIM144 3E

Prepared for:

Sir Airul Syazwan Bin Norshahimi

Submission date:

04 January 2024

"Report in Python Program: Cat Owner Registration"

Prepared by:

Nurul Sofiyyah Binti Azhar (2022810368)

Class: KCDIM144 3E

School of Information Science,

College of Computing, Informatics, and Mathematics,

Universiti Teknologi mara,

Merbok, Kedah.

04 January 2024

Acknowledgment

First of all, I would like to praise and thank the Almighty God for providing me with the strength, as a result of His Blessing, I was able to complete this assignment. Without His Blessing, I could not have completed this assignment plus also with my hard work.

Besides that, I would like to express my gratitude to the lecturer, Sir Airul Syazwan Bin Norshahimi, for helping me with his guidance. He is always there for support and always asks about the assignment's progress that is given. Alhamdulillah, I successfully completed this assignment on time without any hesitation. I would also like to thank him for teaching me in this course until I understood all the learning materials clearly and nicely, although it was also hard enough to understand and practice it.

Last but not least, thank my loving family for their always encouragement and prayers that have kept me going up till now. I hope my effort was worth it and will help me to get good grades for my final results. To sum up, I have learned new things that I never thought that I had a chance to learn before, and I hope my work will get better in the future.
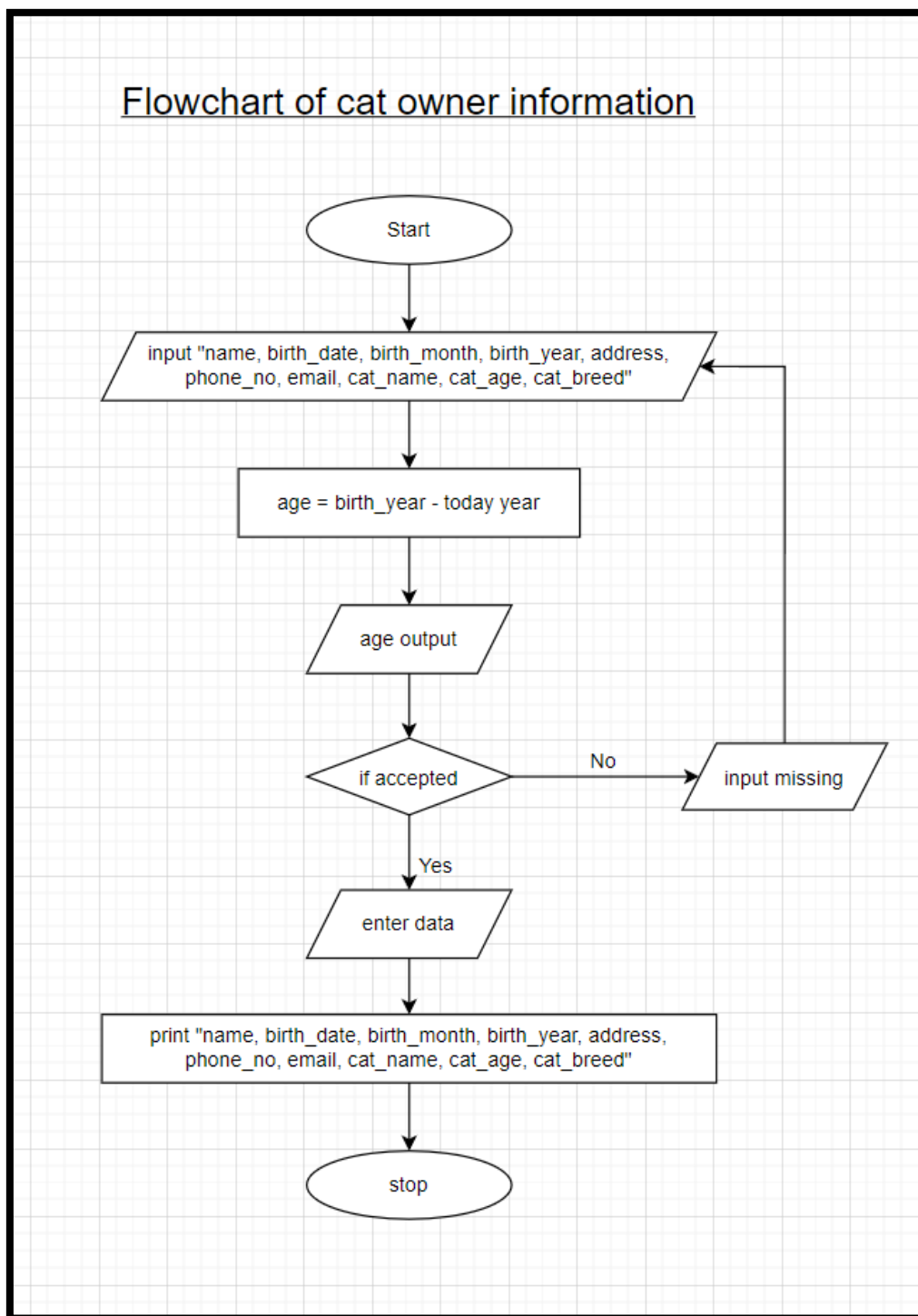
Table of Contents

1.0 Introduction

Many types of hotels that we have nowadays, are not just for us but also have hotels for our pets. In Malaysia, the majority of society chooses a cat as their pet as they have a unique personality and care with their owner. I decided to create a customer registration for a cat owner to board their cat at the cat hotel as my project. The registration needs the owner to fill in their name, birth date, birth month, birth year, address, phone number, email, and also their cat's name, age, and breed. Owner age will be automatically calculated in the registration form, and the owner also needs to accept the terms and conditions before the data can be collected into the database.

2.0 Flowchart.

Flowchart of cat owner information

```
        ( Start )
            |
            v
 /input "name, birth_date, birth_month, birth_year, address,/  <---+
 /      phone_no, email, cat_name, cat_age, cat_breed"   /         |
            |                                                      |
            v                                                      |
 [ age = birth_year - today year ]                                 |
            |                                                      |
            v                                                      |
      / age output /                                               |
            |                                                      |
            v                              No                      |
      < if accepted > -----------------------------> / input missing /
            |
           Yes
            |
            v
      / enter data /
            |
            v
 [ print "name, birth_date, birth_month, birth_year, address, ]
 [       phone_no, email, cat_name, cat_age, cat_breed"      ]
            |
            v
        ( stop )
```

3.0 Snapshot of Code.

```python
# ind_pjc_owner_info.py > ⊗ collect_data
import tkinter as tk
from tkinter import ttk
from tkinter import messagebox
from datetime import date
import mysql.connector

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

mycursor = mydb.cursor()

def collect_data():
    accepted = accept_var.get()
    dt = d_b_entry.get()
    mt = m_b_entry.get()
    yr = y_b_entry.get()
    today = date.today()
    birthdate = date(int(yr), int(mt), int(dt))
    age = today.year - birthdate.year #((today.month, today.day) < (birthdate.month, birthdate.day))

    output_age = tk.Label(cat_owner_frame, text=f"{age} Year",fg="white", bg="#836953")
    output_age.grid(row=4, column=1)

    if accepted == "Accepted":
        name = owner_name_entry.get()
        dbirth = d_b_entry.get()
        mbirth = m_b_entry.get()
        ybirth = y_b_entry.get()
        address = owner_address_entry.get()
        phone = owner_phone_entry.get()
        email = owner_email_address_entry.get()

        if name and address and phone and email:
            cat = cat_name_entry.get()
            cat_age = cat_age_spinbox.get()
            cat_breed = cat_breed_combobox.get()

            print("Name: ", name)
            print("Birth Date:", dbirth, "/", mbirth, "/", ybirth)
            print("Age:", age, "Year")
            print("Address:", address)
            print("Contact No:", phone)
            print("Email:", email)
            print("..............")
            print("Cat Name:", cat)
            print("Age:", cat_age)
            print("Breed:", cat_breed)
            print("----------------------------------------------------------")
        else:
            tk.messagebox.showwarning(title= "Error", message= "There are items that require your attention!")
    else:
        tk.messagebox.showwarning(title= "Error", message= "You have not accepted the terms!")

    sql = "INSERT INTO registration (NAME, AGE, ADDRESS, PHONE_NO, EMAIL, CAT_NAME, CAT_AGE, CAT_BREED) VALUES (%s,%s,%s,%s,%s,%s,%s,%s)"
    val = (name,age, address, phone, email, cat, cat_age, cat_breed)
    mycursor.execute(sql, val)
    mydb.commit()

root = tk.Tk()
root.geometry("520x365")
root['background'] = '#836953'
root.title("Cat Owner Registration")

label = tk.Label(root, text = "Customer Registration", font = ('Comic Sans MS bold', 18), fg="white", bg="#836953")
label.pack(padx=20, pady=1)

frame = tk.Frame(root, bg="#836953")
frame.pack()

#owner
cat_owner_frame = tk.LabelFrame(frame, text = "Personal Detail", fg="white", bg="#836953")
cat_owner_frame.grid(row= 0, column=0, sticky="News", padx=1, pady=1)

owner_name_label = tk.Label(cat_owner_frame, text="Name :", fg="white", bg="#836953")
owner_name_label.grid(row=0, column=0)
owner_name_entry = tk.Entry(cat_owner_frame)
owner_name_entry.grid(row=0, column=1)

owner_birth_date = tk.Label(cat_owner_frame, text="Birth Date :", fg="white", bg="#836953")
owner_birth_date.grid(row=1, column=0)

d_b_entry = ttk.Combobox(cat_owner_frame, values=["1","2","3","4","5","6","7","8","9","10",
                                                  "11","12","13","14","15","16","17","18","19","20",
                                                  "21","22","23","24","25","26","27","28","29","30","31"])
d_b_entry.grid(row=1, column=1)
d_b_entry.set("Select day")
d_b_entry["state"] = 'readonly'

m_b_entry = ttk.Combobox(cat_owner_frame, values=["01","02","03","04","05","06",
                                                  "07","08","09","10","11","12"])
m_b_entry.grid(row=2, column=1)
m_b_entry.set("Select month")
m_b_entry["state"] = 'readonly'

y_b_entry = ttk.Combobox(cat_owner_frame, values=list(range(1960, 2200)))
y_b_entry.grid(row=3, column=1)
y_b_entry.set("select year")
y_b_entry["state"] = 'readonly'

owner_age_label = tk.Label(cat_owner_frame, text="Age :", fg="white", bg="#836953")
owner_age_label.grid(row=4, column=0)
```

```python
107    owner_address_label = tk.Label(cat_owner_frame, text="Address :", fg="white", bg="#836953")
108    owner_address_label.grid(row=5, column=0)
109    owner_address_entry = tk.Entry(cat_owner_frame)
110    owner_address_entry.grid(row=5, column=1)
111
112    owner_phone_label = tk.Label(cat_owner_frame, text="Contact No :", fg="white", bg="#836953")
113    owner_phone_label.grid(row=6, column=0)
114    owner_phone_entry = tk.Entry(cat_owner_frame)
115    owner_phone_entry.grid(row=6, column=1)
116
117    owner_email_address_label = tk.Label(cat_owner_frame, text="Email Address :", fg="white", bg="#836953")
118    owner_email_address_label.grid(row=7, column=0)
119    owner_email_address_entry = tk.Entry(cat_owner_frame)
120    owner_email_address_entry.grid(row=7, column=1)
121
122    for widget in cat_owner_frame.winfo_children():
123        widget.grid_configure(padx= 10, pady=5)
124
125    #cat
126    cat_frame = tk.LabelFrame(frame, text="Cat Details", fg="white", bg="#836953")
127    cat_frame.grid(row=0, column=1, sticky="News", padx=1, pady=1)
128
129    cat_name_label = tk.Label(cat_frame, text="Name :", fg="white", bg="#836953")
130    cat_name_label.grid(row=0, column=0)
131    cat_name_entry = tk.Entry(cat_frame)
132    cat_name_entry.grid(row=0, column=1)
133
134    cat_age_label = tk.Label(cat_frame, text="Age :", fg="white", bg="#836953")
135    cat_age_label.grid(row=1, column=0)
136    cat_age_spinbox = ttk.Combobox(cat_frame, values=["Kitten(0-1)", "Young Adult(1-6)", "Mature Adult(7-10)", "Senior(>10)"])
137    cat_age_spinbox.grid(row=1, column=1)
138    cat_age_spinbox.set("Choose age")
139    cat_age_spinbox["state"] = 'readonly'
140
141    cat_breed_label = tk.Label(cat_frame, text="Breed :", fg="white", bg="#836953")
142    cat_breed_label.grid(row=2, column=0)
143    cat_breed_combobox = ttk.Combobox(cat_frame, values=["Domestic", "American", "Siamese", "Maine Coon", "Ragdoll",
144                                        "Russian Blue", "Bengal", "Bombay", "Persian"])
145    cat_breed_combobox.grid(row=2, column=1)
146    cat_breed_combobox.set("Choose breed")
147    cat_breed_combobox["state"] = 'readonly'
148
149    for widget in cat_frame.winfo_children():
150        widget.grid_configure(padx=10, pady=5)
151
152    #terms
153    terms_frame = tk.LabelFrame(frame, text="Terms & Condition", fg="white", bg="#836953")
154    terms_frame.grid(row=2, column=0, sticky="News", padx=1, pady=1)
155
156    accept_var = tk.StringVar(value="Not Accepted")
157    terms_check = tk.Checkbutton(terms_frame, text="I accept the terms and condition.", fg="#3d251e",
158                                    variable=accept_var, onvalue="Accepted", offvalue="Not Accepted", bg="#836953")
159    terms_check.grid(row=0, column=0)
160
161    #enter button
162    enter_button = tk.Button(frame, text="Enter", bg="white", command= collect_data)
163    enter_button.place(x=325, y=283, width=90, height=30)
164
165    close_button = tk.Button(frame, text="Close", bg="white", command=root.destroy)
166    close_button.place(x=416, y=283, width=90, height=30)
167
168    root.mainloop()
```

4.0 Snapshot of GUI.

5.0 Snapshot of Database.