



**DEPARTMENT OF INFORMATION & COMMUNICATION TECHNOLOGY PROJECT**  
**| SESSION II: 2022/2023**  
**DFP40203 – PYTHON PROGRAMMING**

**CLO3:** Display the ability to visualize the development process of application (P3, PLO6)

<b>NAME :</b>	<b>PUTERA NAQIB KHUSAIRI BIN ASRI</b>
	<b>MOHAMAD ALIFF BIN AZIZAN</b>
<b>REGISTRATION NO.</b>	<b>10DDT21F1021</b>
	<b>10DDT21F1014</b>
<b>CLASS :</b>	<b>DDT5A</b>

**Project instruction**

Work in two- or three-person teams. Discuss the system as a group using one of the topics on the list. Use the project specifications as a guide for creating this project. The following requirements must be met by this project. Present your group project in class.

**List of topic:**

- a) Hotel booking system.
- b) Car rent system.
- c) Purchase system.
- d) Loan system.
- e) Inventory system.

\*\*\* you can choose other topics as long as meets the requirement as stated below.

**Project Requirement:**

**A. CODE**

- Use Object-Oriented Python concept.
- Import and use Modules.
- Define the component of Object-Oriented Python at code. □ Provide comment for each code in the program.

**B. DIRECTORY HANDLING**

- Get the content of a directory.
- Use Raising Exception Handling concept.

**C. GUI**

- Windows, at least three windows.
- Implement e suitable GUI and event in graphical windows.

**D. DATABASE**

- Content at least two tables in a database.
- Basic requirement of database activity (Insert, Update, Delete and Select). Any SQL command to manipulate data like Search.

**E. OUTPUT**

- Must provide at least one receipt or report.

**Report Requirement:**

- a) Title of the system.
- b) Summary of the system.
- c) Objective of the system.
- d) Boundary of system.
- e) Code of the system.
- f) Interface or output of the system.

**A) TITLE OF THE SYSTEM:**

sports martial arts category registration system

**B) SUMMARY OF THE SYSTEM:**

Sports Martial Arts Category Registration System is a software application designed to help competitors register in their categories and to help event organizers manage the registration process for tournaments and competitions. This system will allow competitors to register for events.

**C) OBJECTIVE OT THE SYSTEM:**

To provide a platform for competitors to register in their categories and event organizers to manage tournament registration efficiently and coordinate the competition process.

**D) BOUNDARY OF THE SYSTEM:**

has limitations in the form of an online application that can be accessed through a web browser, including a registration form for participants and a dashboard for event managers, and a reporting module to generate receipts and reports.

## E) CODE OF THE SYSTEM:

SportMartialArtsRegSystem.py:

```

1 import tkinter as prjk
2 from Kategory import ktgory as ctgory
3 from tkinter import messagebox
4 import mysql.connector
5
6 def karte():
7     print("KARATE clicked!\n")
8     try:
9         from Database import KARATE as cdatabase
10        cdatabase.dtaSQL()
11    except:
12        print("Error in Database\n")
13        from SMARS import KARATE as KARATE
14        KARATE.krt()
15
16 def twndo():
17     print("TAEKWONDO clicked!\n")
18     try:
19         from Database import TAEKWONDO as cdatabase
20        cdatabase.dtaSQL()
21    except:
22        print("Error in Database\n")
23        from SMARS import TAEKWONDO as TAEKWONDO
24        TAEKWONDO.taekwondo()
25
26 def myt():
27     print("MUAYTHAI clicked!\n")
28     try:
29         from Database import MUAYTHAI as cdatabase
30        cdatabase.dtaSQL()
31    except:
32        print("Error in Database\n")
33        from SMARS import MUAYTHAI as MUAYTHAI
34        MUAYTHAI.mt()
35
36 def silt():
37     print("SILAT clicked!\n")
38     try:
39         from Database import SILAT as cdatabase
40        cdatabase.dtaSQL()
41    except:
42        print("Error in Database\n")
43        from SMARS import SILAT as SILAT
44        SILAT.silat()
45
46 def bxng():
47     print("BOXING clicked!\n")
48     try:
49         from Database import BOXING as cdatabase
50        cdatabase.dtaSQL()
51    except:
52        print("Error in Database\n")
53        from SMARS import BOXING as BOXING
54        BOXING.bx()
55
56 def mma():
57     print("MMA clicked!\n")
58     try:
59         from Database import MMA as cdatabase
60        cdatabase.dtaSQL()
61    except:
62        print("Error in Database\n")
63        from SMARS import MMA as MMA
64        MMA.mma()
65
66 projekwnds = prjk.Tk()
67 projekwnds.geometry("400x650")
68 projekwnds.title("Sport Martial Arts Registration System")
69
70 canvas = prjk.Canvas(projekwnds, width=400, height=630)
71 canvas.pack()
72 bground = prjk.PhotoImage(file="img_gif/bg.gif")
73 canvas.create_image(0, 0, anchor="nw", image=bground)
74
75 wlcm = prjk.Label(canvas, text="***Welcome To Sport Martial Arts Registration***\n"+
76 "Please Choose Your Sport Martial Arts :-", #Create Label
#Use tkinter library for Python GUI
#Import term in ktgory.py from Kategory file
#Import messagebox from tkinter library
#Imports MySQL Connector

#karte function
#print text
#try to do,
#Import term in KARATE.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in KARATE.py from SMARS file
#call krt funtion from KARATE

#twndo function
#print text
#try to do,
#Import term in TAEKWONDO.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in TAEKWONDO.py from SMARS file
#call taekwondo funtion from TAEKWONDO

#myt function
#print text
#try to do,
#Import term in MUAYTHAI.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in MUAYTHAI.py from SMARS file
#call mt funtion from MUAYTHAI

#silt function
#print text
#try to do,
#Import term in SILAT.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in SILAT.py from SMARS file
#call silat funtion from SILAT

#bxng function
#print text
#try to do,
#Import term in BOXING.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in BOXING.py from SMARS file
#call bx funtion from BOXING

#mma function
#print text
#try to do,
#Import term in MMA.py from Database file
#call function dtaSQL from cdatabase
#if error, do
#print text
#Import term in MMA.py from SMARS file
#call mma funtion from MMA

#Create Tkinter window
#Set winow size = 400 X 650
#Set title

#Create canvas with size = 400 X 650
#Displays the canvas on the screen
#Create PhotoImage use img drom file img_gif
#Create image background use canvas

```

```

77         font=("Times New Roman", 14))
78 wlcm.grid(row=0, column=0, columnspan=2)
79
80 bphoto = prjk.PhotoImage(file="img_gif/boxing.gif")
81 kphoto = prjk.PhotoImage(file="img_gif/karate.gif")
82 mmphoto = prjk.PhotoImage(file="img_gif/mma.gif")
83 mtphoto = prjk.PhotoImage(file="img_gif/muaythai.gif")
84 sphoto = prjk.PhotoImage(file="img_gif/silat.gif")
85 tphoto = prjk.PhotoImage(file="img_gif/taekwando.gif")
86
87 kbtn = prjk.Button(canvas, image=kphoto, command=karte)
88 kbtn.grid(row=1, column=0)
89
90 tbtn = prjk.Button(canvas, image=tpphoto, command=twndo)
91 tbtn.grid(row=1, column=1)
92
93 kbtn = prjk.Button(canvas, text="KARATE",
94                     font=("Times New Roman", 9), command=karte)
95 kbtn.grid(row=2, column=0)
96
97 tbtn = prjk.Button(canvas, text="TAEKWONDO",
98                     font=("Times New Roman", 9), command=twndo)
99 tbtn.grid(row=2, column=1)
100
101 mtbtn= prjk.Button(canvas, image=mtphoto, command=mt)
102 mtbtn.grid(row=3, column=0)
103
104 sbtn = prjk.Button(canvas, image=sphoto, command=slt)
105 sbtn.grid(row=3, column=1)
106
107 mtbtn = prjk.Button(canvas, text="MUAYTHAI",
108                      font=("Times New Roman", 9), command=mt)
109 mtbtn.grid(row=4, column=0)
110
111 sbtn = prjk.Button(canvas, text="SILAT",
112                      font=("Times New Roman", 9), command=slt)
113 sbtn.grid(row=4, column=1)
114
115 bbtn= prjk.Button(canvas, image=bphoto, command=bxng)
116 bbtn.grid(row=5, column=0)
117
118 mmbtn = prjk.Button(canvas, image=mmphoto, command=mMa)
119 mmbtn.grid(row=5, column=1)
120
121 bbtn = prjk.Button(canvas, text="BOXING",
122                      font=("Times New Roman", 9), command=bxng)
123 bbtn.grid(row=6, column=0)
124
125 mmbtn = prjk.Button(canvas, text="MMA",
126                      font=("Times New Roman", 9), command=mMa)
127 mmbtn.grid(row=6, column=1)
128
129 projekwnds.mainloop()

```

## Kategorie/Ktgory.py:

```

1 def Krt(Weight):
2     if Weight <=55:
3         category="55kg"
4     elif Weight >=56 and Weight<=60:
5         category="60kg"
6     elif Weight >=61 and Weight<=70:
7         category="70kg"
8     elif Weight >=71 and Weight<=75:
9         category="75kg"
10    elif Weight >=76 and Weight<=80:
11        category="80kg"
12    else:
13        category="+80kg"
14
15    return category
16
17 def Twdo(Weight):
18     if Weight <54:
19         category="Fin (-54kg)"
20     elif Weight >=54 and Weight<=58:
21         category="Fly (54kg - 58kg)"
22     elif Weight >=58 and Weight<=62:
23         category="Bantam (59kg - 62kg)"
24     elif Weight >=62 and Weight<=67:
25         category="Feather (63kg - 67kg)"
26     elif Weight >=69 and Weight<=72:
27         category="Light (68kg - 72kg)"
28     elif Weight >=72 and Weight<=78:
29         category="Welter (73kg - 78kg)"
30     elif Weight >=78 and Weight<=84:
31         category="Middle (79kg - 84kg)"
32     else:
33         category="Heavy (+84kg)"
34
35    return category
36
37 def Mythai(Weight):
38     if Weight >74:
39         category="Heavyweight (+74kg)"
40     elif Weight >=68 and Weight<=73:
41         category="Middleweight (68kg - 73kg)"
42     elif Weight >=64 and Weight<=67:
43         category="Welterweight (64kg - 67kg)"
44     elif Weight >=59 and Weight<=63:
45         category="Lightweight (59kg - 63kg)"
46     elif Weight >=55 and Weight<=58:
47         category="Featherweight (55kg - 58kg)"
48     elif Weight >=52 and Weight<=54:
49         category="Bantamweight (52kg - 54kg)"
50     elif Weight >=49 and Weight<=51:
51         category="Flyweight (49kg - 51kg)"
52     else:
53         category="Mini Flyweigh (-48kg)"
54
55    return category
56
57 def Slt(Weight):
58     if Weight <=49:
59         category="Class A (-49kg)"
60     elif Weight >=50 and Weight<=55:
61         category="Class B (50kg - 55kg)"
62     elif Weight >=56 and Weight<=60:
63         category="Class C (56kg - 60kg)"
64     elif Weight >=61 and Weight<=65:
65         category="Class D (61kg - 65kg)"
66     elif Weight >=66 and Weight<=70:
67         category="Class E (66kg - 70kg)"
68     elif Weight >=71 and Weight<=75:
69         category="Class F (71kg - 75kg)"
70     elif Weight >=76 and Weight<=80:
71         category="Class G (76kg - 80kg)"
72     elif Weight >=81 and Weight<=85:
73         category="Class H (81kg - 85kg)"
74     elif Weight >=86 and Weight<=90:
75         category="Class I (86kg - 90kg)"
76     elif Weight >=91 and Weight<=95:
77         category="Class J (91kg - 95kg)"
78
79    return category
80
81 #Krt function with variable Weight
82 #if Weight <=55 , do
83 #Set category = 55kg
84 #if Weight 56 - 60 , do
85 #Set category = 60kg
86 #if Weight 61 - 70 , do
87 #Set category = 70kg
88 #if Weight 71 - 75 , do
89 #Set category = 75kg
90 #if Weight 76 - 80 , do
91 #Set category = 80kg
92 #if Weight >80 , do
93 #Set category = +80kg
94
95 #return category value
96
97 #Twdo function with variable Weight
98 #if Weight <54 , do
99 #Set category = Fin (-54kg)
100 #if Weight 54 - 58 , do
101 #Set category = Fly (54kg - 58kg)
102 #if Weight 58 - 62 , do
103 #Set category = Bantam (59kg - 62kg)
104 #if Weight 62 - 67 , do
105 #Set category = Feather (63kg - 67kg)
106 #if Weight 69 - 72 , do
107 #Set category = Light (68kg - 72kg)
108 #if Weight 72 - 78 , do
109 #Set category = Welter (73kg - 78kg)
110 #if Weight 78 - 84 , do
111 #Set category = Middle (79kg - 84kg)
112 #if Weight >84 , do
113 #Set category = Heavy (+84kg)
114
115 #return category value
116
117 #Mythai function with variable Weight
118 #if Weight >74 , do
119 #Set category = Heavyweight (+74kg)
120 #if Weight 68 - 73 , do
121 #Set category = Middleweight (68kg - 73kg)
122 #if Weight 64 - 67 , do
123 #Set category = Welterweight (64kg - 67kg)
124 #if Weight 59 - 63 , do
125 #Set category = Lightweight (59kg - 63kg)
126 #if Weight 55 - 58 , do
127 #Set category = Featherweight (55kg - 58kg)
128 #if Weight 52 - 54 , do
129 #Set category = Bantamweight (52kg - 54kg)
130 #if Weight 49 - 51 , do
131 #Set category = Flyweight (49kg - 51kg)
132 #if Weight >49 , do
133 #Set category = Mini Flyweigh (-48kg)
134
135 #return category value
136
137 #Slt function with variable Weight
138 #if Weight <=49 , do
139 #Set category = Class A (-49kg)
140 #if Weight 50 - 55 , do
141 #Set category = Class B (50kg - 55kg)
142 #if Weight 56 - 60 , do
143 #Set category = Class C (56kg - 60kg)
144 #if Weight 61 - 65 , do
145 #Set category = Class D (61kg - 65kg)
146 #if Weight 66 - 70 , do
147 #Set category = Class E (66kg - 70kg)
148 #if Weight 71 - 75 , do
149 #Set category = Class F (71kg - 75kg)
150 #if Weight 76 - 80 , do
151 #Set category = Class G (76kg - 80kg)
152 #if Weight 81 - 85 , do
153 #Set category = Class H (81kg - 85kg)
154 #if Weight 86 - 90 , do
155 #Set category = Class I (86kg - 90kg)
156 #if Weight 91 - 95 , do
157
158 #return category value

```

```

77         category="Class J (91kg - 95kg)"
78     else:
79         category="Class Open (+96kg)"
80
81     return category
82
83 def Bxg(Weight):
84     if Weight >=91:
85         category="Heavyweight (+91kg)"
86     elif Weight >=81 and Weight<=90:
87         category="Cruiserweight (81kg - 90kg)"
88     elif Weight >=78 and Weight<=80:
89         category="heavyweight (78kg - 80kg)"
90     elif Weight >=67 and Weight<=76:
91         category="middleweight (67kg - 76kg)"
92     else:
93         category="Welterweight (-66kg)"
94
95     return category
96
97 def MMA(Weight):
98     if Weight<=52:
99         category="Strawweight (-52kg)"
100    elif Weight>=53 and Weight<=57:
101        category="Flyweight (53kg - 57kg)"
102    elif Weight>=58 and Weight<=61:
103        category="Bantamweight (58kg - 61kg)"
104    elif Weight>=62 and Weight<=66:
105        category="Featherweight (62kg - 66kg)"
106    elif Weight>=67 and Weight<=70:
107        category="Lightweight (67kg - 70kg)"
108    elif Weight>=71 and Weight<=77:
109        category="Welterweight (71kg - 77kg)"
110    elif Weight>=78 and Weight<=89:
111        category="Middleweight (78kg - 89kg)"
112    elif Weight>=90 and Weight<=93:
113        category="Light Heavyweight (90kg - 93kg)"
114    elif Weight>=94 and Weight<=120:
115        category="Heavyweight (94kg - 120kg)"
116    else:
117        category="Super Heavyweight (+121kg)"
118
119     return category

```

### Database/ BOXING.py:

```
1 import mysql.connector                                         #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                           #dtaSQL funtion
5         host="localhost",                                     #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                            #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exixts
11
12 try:                                                       #try to do
13     projectdatabase = mysql.connector.connect(           #connect to abc_company database
14         host="localhost",
15         user="root",
16         password="",
17         database="smars")
18
19     mydbse = projectdatabase.cursor()                   #Instantiating a cursor for mydbse
20     mydbse.execute("CREATE TABLE IF NOT EXISTS boxing "  #Create table boxing if not exixts
21         "(nama VARCHAR(200), "
22         "age INT, nric VARCHAR(12), gender VARCHAR(6), "
23         "email VARCHAR(200), "
24         "phonenum VARCHAR(15), "
25         "weight VARCHAR(100), "
26         "category VARCHAR(100))")
27 except mysql.connector.Error as err:                  #if error , do
28     print("Error creating table: {}".format(err))      #print error text
29
```

### Database/ KARATE.py:

```
1 import mysql.connector                                         #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                           #dtaSQL funtion
5         host="localhost",                                     #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                            #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exixts
11
12 try:                                                       #try to do
13     projectdatabase = mysql.connector.connect(           #connect to abc_company database
14         host="localhost",
15         user="root",
16         password="",
17         database="smars")
18
19     mydbse = projectdatabase.cursor()                   #Instantiating a cursor for mydbse
20     mydbse.execute("CREATE TABLE IF NOT EXISTS karate "  #Create table karate if not exixts
21         "(nama VARCHAR(200), "
22         "age INT, nric VARCHAR(12), gender VARCHAR(6), "
23         "email VARCHAR(200), "
24         "phonenum VARCHAR(15), "
25         "weight VARCHAR(100), "
26         "category VARCHAR(100))")
27 except mysql.connector.Error as err:                  #if error , do
28     print("Error creating table: {}".format(err))      #print error text
29
```

### Database/ MMA.py:

```
1 import mysql.connector                                         #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                           #dtaSQL funtion
5         host="localhost",                                     #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                            #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exixts
11
12 try:                                                       #try to do
13     projectdatabase = mysql.connector.connect(           #connect to abc_company database
14         host="localhost",
15         user="root",
16         password="",
17         database="smars")
18
19     mydbse = projectdatabase.cursor()                   #Instantiating a cursor for mydbse
20     mydbse.execute("CREATE TABLE IF NOT EXISTS mma "    #Create table mma if not exixts
21         "(nama VARCHAR(200), "
22         "age INT, nric VARCHAR(12), gender VARCHAR(6), "
23         "email VARCHAR(200), "
24         "phonenum VARCHAR(15), "
25         "weight VARCHAR(100), "
26         "category VARCHAR(100))")
27 except mysql.connector.Error as err:                  #if error , do
28     print("Error creating table: {}".format(err))      #print error text
29
```

### Database/ MUAYTHAI.py:

```
1 import mysql.connector                                     #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                      #dtaSQL funtion
5         host="localhost",                                #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                         #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exists
11
12    try:                                              #try to do
13        projectdatabase = mysql.connector.connect(      #connect to abc_company database
14            host="localhost",
15            user="root",
16            password="",
17            database="smars")
18
19        mydbse = projectdatabase.cursor()                #Instantiating a cursor for mydbse
20        mydbse.execute("CREATE TABLE IF NOT EXISTS muaythai " #Create table muaythai if not exists
21            "(nama VARCHAR(200), "
22            "age INT, nric VARCHAR(12), gender VARCHAR(6) ,"
23            "email VARCHAR(200), "
24            "phonenum VARCHAR(15), "
25            "weight VARCHAR(100), "
26            "category VARCHAR(100))")
27    except mysql.connector.Error as err:               #if error , do
28        print("Error creating table: {}".format(err))  #print error text
29
```

### Database/ SILAT.py:

```
1 import mysql.connector                                     #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                      #dtaSQL funtion
5         host="localhost",                                #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                         #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exists
11
12    try:                                              #try to do
13        projectdatabase = mysql.connector.connect(      #connect to abc_company database
14            host="localhost",
15            user="root",
16            password="",
17            database="smars")
18
19        mydbse = projectdatabase.cursor()                #Instantiating a cursor for mydbse
20        mydbse.execute("CREATE TABLE IF NOT EXISTS silat " #Create table silat if not exists
21            "(nama VARCHAR(200), "
22            "age INT, nric VARCHAR(12), gender VARCHAR(6) ,"
23            "email VARCHAR(200), "
24            "phonenum VARCHAR(15), "
25            "weight VARCHAR(100), "
26            "category VARCHAR(100))")
27    except mysql.connector.Error as err:               #if error , do
28        print("Error creating table: {}".format(err))  #print error text
29
```

### Database/ TAEKWONDO.py:

```
1 import mysql.connector                                     #Imports MySQL Connector
2
3 def dtaSQL():
4     mydb = mysql.connector.connect(                      #dtaSQL funtion
5         host="localhost",                                #connect to database
6         user="root",
7         password="")
8
9     myprojectdb = mydb.cursor()                         #Instantiating a cursor for myprojectdb
10    myprojectdb.execute("CREATE DATABASE IF NOT EXISTS smars") #Create database if not exists
11
12    try:                                              #try to do
13        projectdatabase = mysql.connector.connect(      #connect to abc_company database
14            host="localhost",
15            user="root",
16            password="",
17            database="smars")
18
19        mydbse = projectdatabase.cursor()                #Instantiating a cursor for mydbse
20        mydbse.execute("CREATE TABLE IF NOT EXISTS taekwondo " #Create table taekwondo if not exists
21            "(nama VARCHAR(200), "
22            "age INT, nric VARCHAR(12), gender VARCHAR(6) ,"
23            "email VARCHAR(200), "
24            "phonenum VARCHAR(15), "
25            "weight VARCHAR(100), "
26            "category VARCHAR(100))")
27    except mysql.connector.Error as err:               #if error , do
28        print("Error creating table: {}".format(err))  #print error text
29
```

## SMARS / BOXING.py:

```

112     rstwnds.title("Your Registration Receipt")
113
114     rstrflbl=prjk.Label(rstwnds,text="\n *** Receipt ***\n"+
115         " *** Sports Martial Arts : "+SMA+" ***\n",
116         font=("Times New Roman",13))
117     rstrflbl.grid(row=0,column=0)
118
119     DataFrame=prjk.LabelFrame(rstwnds,text="Your Information : ",
120         padx=10,pady=10)
121     DataFrame.grid(row=1,column=0,padx=10)
122
123     nLbl=prjk.Label(DataFrame,text="NAME")
124     nLbl.grid(row=0,column=0,padx=5,pady=5)
125     nDot=prjk.Label(DataFrame,text=" : ")
126     nDot.grid(row=0,column=1,padx=5,pady=5)
127     n=prjk.Entry(DataFrame)
128     n.insert(prjk.END,Name)
129     n.config(state='disabled')
130     n.grid(row=0,column=2,padx=5,pady=5)
131
132     aLbl=prjk.Label(DataFrame,text="AGE")
133     aLbl.grid(row=1,column=0,padx=5,pady=5)
134     aDot=prjk.Label(DataFrame,text=" : ")
135     aDot.grid(row=1,column=1,padx=5,pady=5)
136     a=prjk.Entry(DataFrame)
137     a.insert(prjk.END,age)
138     a.config(state='disabled')
139     a.grid(row=1,column=2,padx=5,pady=5)
140
141     gLbl=prjk.Label(DataFrame,text="GENDER")
142     gLbl.grid(row=2,column=0,padx=5,pady=5)
143     gDot=prjk.Label(DataFrame,text=" : ")
144     gDot.grid(row=2,column=1,padx=5,pady=5)
145     g=prjk.Entry(DataFrame)
146     g.insert(prjk.END,gend)
147     g.config(state='disabled')
148     g.grid(row=2,column=2,padx=5,pady=5)
149
150     nLbl=prjk.Label(DataFrame,text="NRIC")
151     nLbl.grid(row=2,column=0,padx=5,pady=5)
152     nDot=prjk.Label(DataFrame,text=" : ")
153     nDot.grid(row=2,column=1,padx=5,pady=5)
154     n=prjk.Entry(DataFrame)
155     n.insert(prjk.END,str(ic))
156     n.config(state='disabled')
157     n.grid(row=2,column=2,padx=5,pady=5)
158
159     eLbl=prjk.Label(DataFrame,text="E-MAIL")
160     eLbl.grid(row=3,column=0,padx=5,pady=5)
161     eDot=prjk.Label(DataFrame,text=" : ")
162     eDot.grid(row=3,column=1,padx=5,pady=5)
163     e=prjk.Entry(DataFrame)
164     e.insert(prjk.END,Mail)
165     e.config(state='disabled')
166     e.grid(row=3,column=2,padx=5,pady=5)
167
168     pLbl=prjk.Label(DataFrame,text="PHONE NUMBER")
169     pLbl.grid(row=4,column=0,padx=5,pady=5)
170     pDot=prjk.Label(DataFrame,text=" : ")
171     pDot.grid(row=4,column=1,padx=5,pady=5)
172     p=prjk.Entry(DataFrame)
173     p.insert(prjk.END,PhoneNum)
174     p.config(state='disabled')
175     p.grid(row=4,column=2,padx=5,pady=5)
176
177     wLbl=prjk.Label(DataFrame,text="WEIGHT")
178     wLbl.grid(row=5,column=0,padx=5,pady=5)
179     wDot=prjk.Label(DataFrame,text=" : ")
180     wDot.grid(row=5,column=1,padx=5,pady=5)
181     w=prjk.Entry(DataFrame)
182     w.insert(prjk.END,Weight)
183     w.config(state='disabled')
184     w.grid(row=5,column=2,padx=5,pady=5)
185
186     cLbl=prjk.Label(DataFrame,text="CATEGORY")
187     cLbl.grid(row=6,column=0,padx=5,pady=5)
188     cDot=prjk.Label(DataFrame,text=" : ")
189     cDot.grid(row=6,column=1,padx=5,pady=5)
190     c=prjk.Entry(DataFrame)
191     c.insert(prjk.END,category)
192     c.config(state='disabled')
193     c.grid(row=6,column=2,padx=5,pady=5)
194
195     rstwnds.mainloop()
196     print("You Successfully Registered in "+SMA+"\n")
197     messagebox.showinfo("Successfully","Your data inserted successfully")
198     confwnds.withdraw()
199     wnds.withdraw()
200     printrest()
201     resit()
202
203     print("Please Confirm Your Data\n")
204     confwnds = prjk.Tk()
205     confwnds.geometry("375x450")
206     confwnds.title("Confirm Your Data")
207     confwnds.configure(bg="#faf8f5")
208
209     confrmlbl=prjk.Label(confwnds,text="\n *** Please Confirm Your Data ***\n"+
210         " Sports Martial Arts : Boxing\n",
211         bg="#faf8f5",font=("Times New Roman",18))
212     confrmlbl.grid(row=0,column=0)
213
214     DataFrame=prjk.LabelFrame(confwnds,text="Your Information : ",
215         padx=10,pady=10,bg="#faf8f5")
216     DataFrame.grid(row=1,column=0,padx=10)
217
218     nLbl=prjk.Label(DataFrame,text="NAME", bg="#faf8f5")
219     nLbl.grid(row=0,column=0,padx=5,pady=5)
220     nDot=prjk.Label(DataFrame,text=" : ", bg="#faf8f5")
221     nDot.grid(row=0,column=1,padx=5,pady=5)
222     n=prjk.Entry(DataFrame)

```



```

334         bg="#faf8f5",font=("Times New Roman",14))
335         wldm.grid(row=0,column=0)
336
337 ▼ FillFrm=prjk.LabelFrame(wnd, text="Fill Your Information Here : ",
338     padx=10, pady=10, bg="#faf8f5")
339     FillFrm.grid(row=1,column=0,padx=10)
340
341     nameLbl=prjk.Label(FillFrm,text="NAME", bg="#faf8f5")
342     nameLbl.grid(row=0, column=0, padx=5, pady=5)
343     nameDot=prjk.Label(FillFrm,text=" : ", bg="#faf8f5")
344     nameDot.grid(row=0, column=1, padx=5, pady=5)
345     name=prjk.Entry(FillFrm)
346     name.grid(row=0, column=2, padx=5, pady=5)
347
348     NrIclbl=prjk.Label(FillFrm, text="NRIC", bg="#faf8f5")
349     NrIclbl.grid(row=1, column=0, padx=5, pady=5)
350     NrIcDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
351     NrIcDot.grid(row=1, column=1, padx=5, pady=5)
352     NrIc=prjk.Entry(FillFrm)
353     NrIc.grid(row=1, column=2, padx=5, pady=5)
354 ▼ NrIcx=prjk.Label(FillFrm, text="Example : 010203040506",
355     bg="#faf8f5", font=("Times New Roman",9))
356     NrIcx.grid(row=1, column=3, padx=5, pady=5)
357
358     EMaillbl=prjk.Label(FillFrm, text="E-MAIL", bg="#faf8f5")
359     EMaillbl.grid(row=2, column=0, padx=5, pady=5)
360     EMailDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
361     EMailDot.grid(row=2, column=1, padx=5, pady=5)
362     EMail=prjk.Entry(FillFrm)
363     EMail.grid(row=2, column=2, padx=5, pady=5)
364
365     PhoneLbl=prjk.Label(FillFrm, text="PHONE NUMBER", bg="#faf8f5")
366     PhoneLbl.grid(row=3, column=0, padx=5, pady=5)
367     PhoneDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
368     PhoneDot.grid(row=3, column=1, padx=5, pady=5)
369     Phone=prjk.Entry(FillFrm)
370     Phone.grid(row=3, column=2, padx=5, pady=5)
371
372     BrtLbl=prjk.Label(FillFrm, text="WEIGHT", bg="#faf8f5")
373     BrtLbl.grid(row=4, column=0, padx=5, pady=5)
374     BrtDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
375     BrtDot.grid(row=4, column=1, padx=5, pady=5)
376     Brt=prjk.Entry(FillFrm)
377     Brt.grid(row=4, column=2, padx=5, pady=5)
378
379     Btnfrm=prjk.Frame(wnd, padx=10, bg="#faf8f5")
380     Btnfrm.grid(row=5, column=0, padx=10, pady=10)
381
382     submitBtn=prjk.Button(Btnfrm, text="SUBMIT", command=bxngsbmt)
383     submitBtn.grid(row=0, column=0, padx=5)
384
385     rstBtn=prjk.Button(Btnfrm, text="RESET", command=bxngrst)
386     rstBtn.grid(row=0, column=1, padx=5)
387
388     Outputfrm=prjk.LabelFrame(wnd,
389     text="", borderwidth=0,
390     padx=10, pady=10, bg="#faf8f5")
391     Outputfrm.grid(row=6, column=0, padx=10)
392
393     ktgotyLbl=prjk.Label(Outputfrm, text="",
394     bg="#faf8f5")
395     ktgotyLbl.grid(row=0, column=0, padx=5, pady=5)
396     ktgotyDot=prjk.Label(Outputfrm, text="",
397     bg="#faf8f5")
398     ktgotyDot.grid(row=0, column=1, padx=5, pady=5)
399     ktgoty=prjk.Label(Outputfrm, text="",
400     bg="#faf8f5")
401     ktgoty.grid(row=0, column=2, padx=5, pady=5)
402
403     wnds.mainloop()
404

```

## SMARS/ KARATE.py:

```

1 import tkinter as prjk
2 from Kategory import ktgory as ctgory
3 from tkinter import messagebox
4 import mysql.connector
5
6 def krt():
7     def krtrst():
8         name.delete(0,prjk.END)
9         NrIc.delete(0,prjk.END)
10        EMail.delete(0,prjk.END)
11        Phone.delete(0,prjk.END)
12        Brt.delete(0,prjk.END)
13        OutputFrm["text"]=""
14        ktgotyLbl["text"]=""
15        ktgotyDot["text"]=""
16        ktgoty["text"]=""
17        OutputFrm.config(borderwidth=0)
18
19 def krtsbmt():
20     OutputFrm["text"]="Your Category : "
21     OutputFrm.config(borderwidth=1)
22     ktgotyLbl["text"]="Category"
23     ktgotyDot["text"]:""
24     Nama=name.get()
25     Ic=NrIc.get()
26     Mail=EMail.get()
27     PhoneNum=Phone.get()
28     Weight=float(Brt.get())
29 try:
30     Nama>Nama.upper()
31     year=int(Ic[0:2])
32     if year>0 and year<22:
33         year+=2000
34     else:
35         year+=1900
36     age=2023-year
37
38     jntina=int(Ic)
39     jntina%2
40     if jntina == 0:
41         gend="FEMALE"
42     else:
43         gend="MALE"
44
45     if Weight>1000:
46         Weight/=1000
47     else:
48         Weight=Weight
49
50     category=ctgory.Krt(Weight)
51     Weight="{:.2f}".format(Weight)
52     Weight=Weight+" kg"
53     Ic=str(Ic)
54     age=str(age)
55     print("NAME      : "+Nama)
56     print("AGE       : "+age)
57     print("GENDER    : "+gend)
58     print("NRIC      : "+Ic)
59     print("EMAIL     : "+Mail)
60     print("PHONE NUMBER : "+PhoneNum)
61     print("WEIGHT    : "+Weight)
62     print("CATEGORY   : "+category)
63     ktgoty["text"]+category
64     print()
65 except:
66     Weight="Invalid"
67     print("Weight "+Weight)
68     OutputFrm["text"]=""
69     ktgotyLbl["text"]=""
70     ktgotyDot["text"]=""
71     ktgoty["text"]=""
72     OutputFrm.config(borderwidth=0)
73
74 def cncl():
75     try:
76         projectdatabase = mysql.connector.connect(
77             host="localhost",
78             user="root",
79             password="",
80             database="smars")
81
82         mydbse = projectdatabase.cursor()
83         mydbse.execute("DELETE FROM karate WHERE nric=%s",
84                         (Ic,))
85         projectdatabase.commit()
86         print(f"Your Registration Canceled")
87
88     except mysql.connector.Error as err:
89         print("Failed delete record : {}".format(err)+"\n")
90
91     messagebox.showinfo("Canceled","You canceled your Registration")
92     confwnds.withdraw()
93     wnds.withdraw()
94
95 def comfrm():
96     def printrest():
97         print("\n\n      **** Recipt ****\n"+
98             "***** Sports Martial Arts : "+SMA+" *****\n")
99         print("\tNAME      : "+Nama)
100        print("\tAGE       : "+age)
101        print("\tGENDER    : "+gend)
102        print("\tNRIC      : "+Ic)
103        print("\tEMAIL     : "+Mail)
104        print("\tPHONE NUMBER : "+PhoneNum)
105        print("\tWEIGHT    : "+Weight)
106        print("\tCATEGORY   : "+category)
107        print("\n*****\n")
108
109     def resit():
110         rswndws = prjk.Tk()
111         rswndws.geometry("315x420")

```

```

112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222

rstwnds.title("Your Registration Receipt")
rstfrmLBL=prjk.Label(rstwnds,text="\n *** Receipt ***\n"+ " *** Sports Martial Arts : "+SMA+" ***\n", font=("Times New Roman",13))
rstfrmLBL.grid(row=0,column=0)

DataFrame=prjk.LabelFrame(rstwnds,text="Your Information : ", padx=10,pady=10)
DataFrame.grid(row=1,column=0,padx=10)

nLbl=prjk.Label(DataFrame,text="NAME")
nLbl.grid(row=0,column=0,padx=5,pady=5)
nDot=prjk.Label(DataFrame,text=" : ")
nDot.grid(row=0,column=1,padx=5,pady=5)
n=prjk.Entry(DataFrame)
n.insert(prjk.END>Nama)
n.config(state='disabled')
n.grid(row=0,column=2,padx=5,pady=5)

aLbl=prjk.Label(DataFrame,text="AGE")
aLbl.grid(row=1,column=0,padx=5,pady=5)
aDot=prjk.Label(DataFrame,text=" : ")
aDot.grid(row=1,column=1,padx=5,pady=5)
a=prjk.Entry(DataFrame)
a.insert(prjk.END,age)
a.config(state='disabled')
a.grid(row=1,column=2,padx=5,pady=5)

gLbl=prjk.Label(DataFrame,text="GENDER")
gLbl.grid(row=2,column=0,padx=5,pady=5)
gDot=prjk.Label(DataFrame,text=" : ")
gDot.grid(row=2,column=1,padx=5,pady=5)
g=prjk.Entry(DataFrame)
g.insert(prjk.END,gend)
g.config(state='disabled')
g.grid(row=2,column=2,padx=5,pady=5)

nLbl=prjk.Label(DataFrame,text="NRIC")
nLbl.grid(row=2,column=0,padx=5,pady=5)
nDot=prjk.Label(DataFrame,text=" : ")
nDot.grid(row=2,column=1,padx=5,pady=5)
n=prjk.Entry(DataFrame)
n.insert(prjk.END,str(Ic))
n.config(state='disabled')
n.grid(row=2,column=2,padx=5,pady=5)

eLbl=prjk.Label(DataFrame,text="E-MAIL")
eLbl.grid(row=3,column=0,padx=5,pady=5)
eDot=prjk.Label(DataFrame,text=" : ")
eDot.grid(row=3,column=1,padx=5,pady=5)
e=prjk.Entry(DataFrame)
e.insert(prjk.END,Mail)
e.config(state='disabled')
e.grid(row=3,column=2,padx=5,pady=5)

pLbl=prjk.Label(DataFrame,text="PHONE NUMBER")
pLbl.grid(row=4,column=0,padx=5,pady=5)
pDot=prjk.Label(DataFrame,text=" : ")
pDot.grid(row=4,column=1,padx=5,pady=5)
p=prjk.Entry(DataFrame)
p.insert(prjk.END,PhoneNum)
p.config(state='disabled')
p.grid(row=4,column=2,padx=5,pady=5)

wLbl=prjk.Label(DataFrame,text="WEIGHT")
wLbl.grid(row=5,column=0,padx=5,pady=5)
wDot=prjk.Label(DataFrame,text=" : ")
wDot.grid(row=5,column=1,padx=5,pady=5)
w=prjk.Entry(DataFrame)
w.insert(prjk.END,Weight)
w.config(state='disabled')
w.grid(row=5,column=2,padx=5,pady=5)

cLbl=prjk.Label(DataFrame,text="CATEGORY")
cLbl.grid(row=6,column=0,padx=5,pady=5)
cDot=prjk.Label(DataFrame,text=" : ")
cDot.grid(row=6,column=1,padx=5,pady=5)
c=prjk.Entry(DataFrame)
c.insert(prjk.END,category)
c.config(state='disabled')
c.grid(row=6,column=2,padx=5,pady=5)

rstwnds.mainloop()
print("You Successfully Registered in "+SMA+"\n")
messagebox.showinfo("Successfully","Your data inserted successfully")
confwnds.withdraw()
wnds.withdraw()
printrest()
resit()

print("Please Confirm Your Data\n")
confwnds = prjk.Tk()
confwnds.geometry("375x420")
confwnds.title("Confirm Your Data")
confwnds.configure(bg="#faf8f5")

confrmLBL=prjk.Label(confwnds,text="\n *** Please Confirm Your Data ***\n"+ " Sports Martial Arts : Karate\n", bg="#faf8f5",font=("Times New Roman",18))
confrmLBL.grid(row=0,column=0)

DataFrame=prjk.LabelFrame(confwnds,text="Your Information : ", padx=10,pady=10,bg="#faf8f5")
DataFrame.grid(row=1,column=0,padx=10)

nLbl=prjk.Label(DataFrame,text="NAME", bg="#faf8f5")
nLbl.grid(row=0,column=0,padx=5,pady=5)
nDot=prjk.Label(DataFrame,text=" : ", bg="#faf8f5")
nDot.grid(row=0,column=1,padx=5,pady=5)
n=prjk.Entry(DataFrame)

```



```

334         bg="#faf8f5",font=("Times New Roman",14))
335         wlcm.grid(row=0,column=0)
336
337 FillFrm=prjk.LabelFrame(wnds, text="Fill Your Information Here : ",
338     padx=10, pady=10, bg="#faf8f5")
339 FillFrm.grid(row=1,column=0,padx=10)
340
341 nameLabel=prjk.Label(FillFrm,text="NAME", bg="#faf8f5")
342 nameLabel.grid(row=0, column=0, padx=5, pady=5)
343 nameDot=prjk.Label(FillFrm,text=" : ", bg="#faf8f5")
344 nameDot.grid(row=0, column=1, padx=5, pady=5)
345 name=prjk.Entry(FillFrm)
346 name.grid(row=0, column=2, padx=5, pady=5)
347
348 NrIclbl=prjk.Label(FillFrm, text="NRIC", bg="#faf8f5")
349 NrIclbl.grid(row=1, column=0, padx=5, pady=5)
350 NrIcDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
351 NrIcDot.grid(row=1, column=1, padx=5, pady=5)
352 NrIc=prjk.Entry(FillFrm)
353 NrIc.grid(row=1, column=2, padx=5, pady=5)
354 NrIcex=prjk.Label(FillFrm, text="Example : 010203040506",
355     bg="#faf8f5",font=("Times New Roman",9))
356 NrIcex.grid(row=1, column=3, padx=5, pady=5)
357
358 EMaillbl=prjk.Label(FillFrm, text="E-MAIL", bg="#faf8f5")
359 EMaillbl.grid(row=2, column=0, padx=5, pady=5)
360 EMaildot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
361 EMaildot.grid(row=2, column=1, padx=5, pady=5)
362 EMail=prjk.Entry(FillFrm)
363 EMail.grid(row=2, column=2, padx=5, pady=5)
364
365 PhoneLbl=prjk.Label(FillFrm, text="PHONE NUMBER", bg="#faf8f5")
366 PhoneLbl.grid(row=3, column=0, padx=5, pady=5)
367 PhoneDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
368 PhoneDot.grid(row=3, column=1, padx=5, pady=5)
369 Phone=prjk.Entry(FillFrm)
370 Phone.grid(row=3, column=2, padx=5, pady=5)
371
372 BrtLbl=prjk.Label(FillFrm, text="WEIGHT", bg="#faf8f5")
373 BrtLbl.grid(row=4, column=0, padx=5, pady=5)
374 BrtDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
375 BrtDot.grid(row=4, column=1, padx=5, pady=5)
376 Brt=prjk.Entry(FillFrm)
377 Brt.grid(row=4, column=2, padx=5, pady=5)
378
379 BtnFrm=prjk.Frame(wnds, padx=10, bg="#faf8f5")
380 BtnFrm.grid(row=5, column=0, padx=10, pady=10)
381
382 submitBtn=prjk.Button(BtnFrm, text="SUBMIT", command=krtsbmt)
383 submitBtn.grid(row=0, column=0, padx=5)
384
385 rstBtn=prjk.Button(BtnFrm, text="RESET", command=krtrst)
386 rstBtn.grid(row=0, column=1, padx=5)
387
388 OutputFrm=prjk.LabelFrame(wnds,
389     text="", borderwidth=0,
390     padx=10, pady=10, bg="#faf8f5")
391 OutputFrm.grid(row=6, column=0, padx=10)
392
393 ktgotyLbl=prjk.Label(OutputFrm, text="",
394     bg="#faf8f5")
395 ktgotyLbl.grid(row=0, column=0, padx=5, pady=5)
396 ktgotyDot=prjk.Label(OutputFrm, text="",
397     bg="#faf8f5")
398 ktgotyDot.grid(row=0, column=1, padx=5, pady=5)
399 ktgoty=prjk.Label(OutputFrm, text="",
400     bg="#faf8f5")
401 ktgoty.grid(row=0, column=2, padx=5, pady=5)
402
403 wnds.mainloop()

```

## SMARS/ MMA.py:

```

1 import tkinter as prjk
2 from Kategory import ktgory as ctgory
3 from tkinter import messagebox
4 import mysql.connector
5
6 def mma():
7     def mmast():
8         name.delete(0,prjk.END)
9         NrIc.delete(0,prjk.END)
10        EMail.delete(0,prjk.END)
11        Phone.delete(0,prjk.END)
12        Brt.delete(0,prjk.END)
13        OutputFrm["text"]=""
14        ktgotyLbl["text"]=""
15        ktgotyDot["text"]=""
16        ktgoty["text"]=""
17        OutputFrm.config(borderwidth=0)
18
19    def mmasbt():
20        OutputFrm["text"]="Your Category : "
21        OutputFrm.config(borderwidth=1)
22        ktgotyLbl["text"]="Category"
23        ktgotyDot["text"]":"
24        Nama=name.get()
25        Ic=NrIc.get()
26        Mail=EMail.get()
27        PhoneNum=Phone.get()
28        Weight=float(Brt.get())
29        try:
30            Nama>Nama.upper()
31            year=int(Ic[0:2])
32            if year>0 and year<=22:
33                year+=2000
34            else:
35                year+=1900
36            age=2023-year
37
38            jntina=int(Ic)
39            jntina%2
40            if jntina == 0:
41                gend="FEMALE"
42            else:
43                gend="MALE"
44
45            if Weight>1000:
46                Weight/=1000
47            else:
48                Weight=Weight
49
50            category=ctgory.MmA(Weight)
51            Weight=":.2f".format(Weight)
52            Weight=Weight+" kg"
53            Ic=str(Ic)
54            age=str(age)
55            print("NAME      : "+Nama)
56            print("AGE       : "+age)
57            print("GENDER    : "+gend)
58            print("NRIC      : "+Ic)
59            print("EMAIL     : "+Mail)
60            print("PHONE NUMBER : "+PhoneNum)
61            print("WEIGHT    : "+Weight)
62            print("CATEGORY  : "+category)
63            ktgoty["text"]=category
64            print()
65        except:
66            Weight="Invalid"
67            print("Weight "+Weight)
68            OutputFrm["text"]=""
69            ktgotyLbl["text"]=""
70            ktgotyDot["text"]=""
71            ktgoty["text"]=""
72            OutputFrm.config(borderwidth=0)
73
74    def cncl():
75        try:
76            projectdatabase = mysql.connector.connect(
77                host="localhost",
78                user="root",
79                password="",
80                database="smars")
81
82            mydbse = projectdatabase.cursor()
83            mydbse.execute("DELETE FROM mma WHERE nrnic=%s",
84                          (Ic,))
85            projectdatabase.commit()
86            print("Your Registration Canceled")
87
88        except mysql.connector.Error as err:
89            print("Failed delete record : {}".format(err)+"\n")
90
91        messagebox.showinfo("Canceled","You canceled your Registration")
92        confwnds.withdraw()
93        wnds.withdraw()
94
95    def comfrm():
96        def printrest():
97            print("\n\n      ***** Receipt *****\n")
98            print("      NAME      : "+Nama)
99            print("      AGE       : "+age)
100           print("      GENDER    : "+gend)
101           print("      NRIC      : "+Ic)
102           print("      EMAIL     : "+Mail)
103           print("      PHONE NUMBER : "+PhoneNum)
104           print("      WEIGHT    : "+Weight)
105           print("      CATEGORY  : "+category)
106           print("\n*****\n")
107
108        def resit():
109            rswndws = prjk.Tk()
110            rswndws.geometry("315x420")

```

```

112         rswnd.title("Your Registration Receipt")
113
114         rstrmbl=prjk.Label(rswnd,text="\n *** Receipt ***\n"+
115             " *** Sports Martial Arts : "+SMA+" ***\n",
116             font=("Times New Roman",13))
117         rstrmbl.grid(row=0,column=0)
118
119         DataFrame=prjk.LabelFrame(rswnd,text="Your Information : ",
120             padx=10,pady=10)
121         DataFrame.grid(row=1,column=0,padx=10)
122
123         nLbl=prjk.Label(DataFrame,text="NAME")
124         nLbl.grid(row=0,column=0,padx=5,pady=5)
125         nDot=prjk.Label(Dataframe,text=" : ")
126         nDot.grid(row=0,column=1,padx=5,pady=5)
127         n=prjk.Entry(DataFrame)
128         n.insert(prjk.END,Nama)
129         n.config(state='disabled')
130         n.grid(row=0,column=2,padx=5,pady=5)
131
132         aLbl=prjk.Label(DataFrame,text="AGE")
133         aLbl.grid(row=1,column=0,padx=5,pady=5)
134         aDot=prjk.Label(Dataframe,text=" : ")
135         aDot.grid(row=1,column=1,padx=5,pady=5)
136         a=prjk.Entry(Dataframe)
137         a.insert(prjk.END,age)
138         a.config(state='disabled')
139         a.grid(row=1,column=2,padx=5,pady=5)
140
141         gLbl=prjk.Label(DataFrame,text="GENDER")
142         gLbl.grid(row=2,column=0,padx=5,pady=5)
143         gDot=prjk.Label(Dataframe,text=" : ")
144         gDot.grid(row=2,column=1,padx=5,pady=5)
145         g=prjk.Entry(Dataframe)
146         g.insert(prjk.END,gend)
147         g.config(state='disabled')
148         g.grid(row=2,column=2,padx=5,pady=5)
149
150         nLbl=prjk.Label(DataFrame,text="NRIC")
151         nLbl.grid(row=2,column=0,padx=5,pady=5)
152         nDot=prjk.Label(Dataframe,text=" : ")
153         nDot.grid(row=2,column=1,padx=5,pady=5)
154         n=prjk.Entry(Dataframe)
155         n.insert(prjk.END,str(Ic))
156         n.config(state='disabled')
157         n.grid(row=2,column=2,padx=5,pady=5)
158
159         eLbl=prjk.Label(DataFrame,text="E-MAIL")
160         eLbl.grid(row=3,column=0,padx=5,pady=5)
161         eDot=prjk.Label(Dataframe,text=" : ")
162         eDot.grid(row=3,column=1,padx=5,pady=5)
163         e=prjk.Entry(Dataframe)
164         e.insert(prjk.END,Mail)
165         e.config(state='disabled')
166         e.grid(row=3,column=2,padx=5,pady=5)
167
168         pLbl=prjk.Label(Dataframe,text="PHONE NUMBER")
169         pLbl.grid(row=4,column=0,padx=5,pady=5)
170         pDot=prjk.Label(Dataframe,text=" : ")
171         pDot.grid(row=4,column=1,padx=5,pady=5)
172         p=prjk.Entry(Dataframe)
173         p.insert(prjk.END,PhoneNum)
174         p.config(state='disabled')
175         p.grid(row=4,column=2,padx=5,pady=5)
176
177         wLbl=prjk.Label(DataFrame,text="WEIGHT")
178         wLbl.grid(row=5,column=0,padx=5,pady=5)
179         wDot=prjk.Label(Dataframe,text=" : ")
180         wDot.grid(row=5,column=1,padx=5,pady=5)
181         w=prjk.Entry(Dataframe)
182         w.insert(prjk.END,Weight)
183         w.config(state='disabled')
184         w.grid(row=5,column=2,padx=5,pady=5)
185
186         cLbl=prjk.Label(DataFrame,text="CATEGORY")
187         cLbl.grid(row=6,column=0,padx=5,pady=5)
188         cDot=prjk.Label(Dataframe,text=" : ")
189         cDot.grid(row=6,column=1,padx=5,pady=5)
190         c=prjk.Entry(Dataframe)
191         c.insert(prjk.END,category)
192         c.config(state='disabled')
193         c.grid(row=6,column=2,padx=5,pady=5)
194
195         rswnd.mainloop()
196         print("You Successfully Registered in "+SMA+"\n")
197         messagebox.showinfo("Successfully","Your data inserted successfully")
198         confwnds.withdraw()
199         wnds.withdraw()
200         printtest()
201         reset()
202
203         print("Please Confirm Your Data\n")
204         confwnds = prjk.Tk()
205         confwnds.geometry("375x450")
206         confwnds.title("Confirm Your Data")
207         confwnds.configure(bg="#fafaf5")
208
209         confrmbl=prjk.Label(confwnds,text="\n *** Please Confirm Your Data ***\n"+
210             " Sports Martial Arts : MMA\n",
211             bg="#fafaf5",font=("Times New Roman",18))
212         confrmbl.grid(row=0,column=0)
213
214         DataFrame=prjk.LabelFrame(confwnds,text="Your Information : ",
215             padx=10,pady=10,bg="#fafaf5")
216         DataFrame.grid(row=1,column=0,padx=10)
217
218         nLbl=prjk.Label(DataFrame,text="NAME", bg="#fafaf5")
219         nLbl.grid(row=0,column=0,padx=5,pady=5)
220         nDot=prjk.Label(Dataframe,text=" : ", bg="#fafaf5")
221         nDot.grid(row=0,column=1,padx=5,pady=5)
222         n=prjk.Entry(Dataframe)

```



```

334         bg="#faf8f5",font=("Times New Roman",14))
335         wlcm.grid(row=0,column=0)
336
337     FillFrm=prjk.LabelFrame(wnds, text="Fill Your Information Here : ",
338         padx=10, pady=10, bg="#faf8f5")
339     FillFrm.grid(row=1,column=0,padx=10)
340
341     nameLbl=prjk.Label(FillFrm,text="NAME", bg="#faf8f5")
342     nameLbl.grid(row=0,column=0,padx=5,pady=5)
343     nameDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
344     nameDot.grid(row=0,column=1,padx=5,pady=5)
345     name=prjk.Entry(FillFrm)
346     name.grid(row=0,column=2,padx=5,pady=5)
347
348     NrIcLbl=prjk.Label(FillFrm, text="NRIC", bg="#faf8f5")
349     NrIcLbl.grid(row=1,column=0,padx=5,pady=5)
350     NrIdot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
351     NrIdot.grid(row=1,column=1,padx=5,pady=5)
352     NrIc=prjk.Entry(FillFrm)
353     NrIc.grid(row=1,column=2,padx=5,pady=5)
354     NrIcex=prjk.Label(FillFrm, text="Example : 01203040506",
355         bg="#faf8f5",font=("Times New Roman",9))
356     NrIcex.grid(row=1,column=3,padx=5,pady=5)
357
358     EMaillbl=prjk.Label(FillFrm, text="E-MAIL", bg="#faf8f5")
359     EMaillbl.grid(row=2,column=0,padx=5,pady=5)
360     EMaildot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
361     EMaildot.grid(row=2,column=1,padx=5,pady=5)
362     EMail=prjk.Entry(FillFrm)
363     EMail.grid(row=2,column=2,padx=5,pady=5)
364
365     PhoneLbl=prjk.Label(FillFrm, text="PHONE NUMBER", bg="#faf8f5")
366     PhoneLbl.grid(row=3,column=0,padx=5,pady=5)
367     PhoneDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
368     PhoneDot.grid(row=3,column=1,padx=5,pady=5)
369     Phone=prjk.Entry(FillFrm)
370     Phone.grid(row=3,column=2,padx=5,pady=5)
371
372     BrtLbl=prjk.Label(FillFrm, text="WEIGHT", bg="#faf8f5")
373     BrtLbl.grid(row=4,column=0,padx=5,pady=5)
374     BrtDot=prjk.Label(FillFrm, text=" : ", bg="#faf8f5")
375     BrtDot.grid(row=4,column=1,padx=5,pady=5)
376     Brt=prjk.Entry(FillFrm)
377     Brt.grid(row=4,column=2,padx=5,pady=5)
378
379     BtnFrm=prjk.Frame(wnds, padx=10, bg="#faf8f5")
380     BtnFrm.grid(row=5,column=0,padx=10, pady=10)
381
382     submitBtn=prjk.Button(BtnFrm, text="SUBMIT", command=mmasbmt)
383     submitBtn.grid(row=0,column=0,padx=5)
384
385     rstBtn=prjk.Button(BtnFrm, text="RESET", command=mmarst)
386     rstBtn.grid(row=0,column=1,padx=5)
387
388     ▼ OutputFrm=prjk.LabelFrame(wnds,
389         text="", borderwidth=0,
390         padx=10, pady=10, bg="#faf8f5")
391     OutputFrm.grid(row=6,column=0,padx=10)
392
393     ▼ ktgotyLbl=prjk.Label(OutputFrm, text="",
394         bg="#faf8f5")
395     ktgotyLbl.grid(row=0,column=0,padx=5,pady=5)
396     ▼ ktgotyDot=prjk.Label(OutputFrm, text="",
397         bg="#faf8f5")
398     ktgotyDot.grid(row=0,column=1,padx=5,pady=5)
399     ▼ ktgoty=prjk.Label(OutputFrm, text="",
400         bg="#faf8f5")
401     ktgoty.grid(row=0,column=2,padx=5,pady=5)
402
403     wnds.mainloop()

```

```

SMARS/ MUAYTHAI.py:
import tkinter as
prjk
#Use tkinter library for Python GUI
from Kategory import ktgory as
ctgory
in ktgory.py from Kategory file
from tkinter import
messagebox
import messagebox from tkinter library
import
mysql.connector
    #Imports MySQL Connector

def
mt():
    #mt function
    def
mytorst():
    #function mytorst
    name.delete(0,prjk.END)
        #clear input in name Entry
    NrIc.delete(0,prjk.END)
        #clear input in NrIc Entry
    EMail.delete(0,prjk.END)
        #clear input in EMail Entry
    Phone.delete(0,prjk.END)
        #clear input in Phone Entry
    Brt.delete(0,prjk.END)
        #clear input in Brt Entry
    OutputFrm["text"]=""
        #set text in OutputFrm to ""
    ktgotyLbl["text"]=""
        #set text in ktgotyLbl to ""
    ktgotyDot["text"]=""
        #set text in ktgotyDot to ""
    ktgoty["text"]=""
        #set text in ktgoty to ""
    OutputFrm.config(borderwidth=0)
        #set border width to OutputFrm to 0

    def
mytosbmt():
    #function mytosbmt
    OutputFrm["text"]="Your Category :
"
        #set text in OutputFrm
    OutputFrm.config(borderwidth=1)
        #set border width to OutputFrm to 1
    ktgotyLbl["text"]="Category"
        #set text in ktgotyLbl
    ktgotyDot["text"]":"
        #set text in ktgotyDot

```

```

Nama=name.get()
    #get name from value in name Entry
Ic=NrIc.get()
    #get Ic from value in NrIc Entry
Mail=EMail.get()
    #get Mail from value in EMail Entry
PhoneNum=Phone.get()
    #get PhoneNum from value in Phone Entry
Weight=float(Brt.get())
    #get float Weight from value in Brt Entry
try:
    #try to do
    Nama>Nama.upper()
        #change name to upper
year=int(Ic[0:2])
    #get year from Ic parameter 0-2
    if year>=0 and
year<=22:                                              #if year
0-22, do
    year+=2000
        #set year equal to year+2000
else:
    #id year >22 , do
    year+=1900
        #set year equal to year+1900
age=2023-
year
et age = 2023 - year

jntina=int(Ic)
    #get jntina from Ic
jntina%=2
    #calculate jntina = jntina % 2
if jntina == 0:                                         #if
jntina equal to 0, do
    gend="FEMALE"
        #set gend to FEMALE
else:
    #IF jntina not equal to 0, do
    gend="MALE"
        #set gend to MALE

if
Weight>=1000:
    #if Weight > 1000, do
    Weight/=1000
        #Set Weight equal to Weight/1000
else:
    #if Weight < 1000
    Weight=Weight
        #Set Weight = Weight

```

```

        category=ctgory.Mythai(Weight)
            #Set category and call Mythai function from ctgory
        Weight="{:.2f}".format(Weight)
            #Set Weight to 2 decimal point
        Weight=Weight+
    kg"
    Weight to Weight+kg
        Ic=str(Ic)
            #Set Ic to string Ic
        age=str(age)
            #Set age to string age
        print("NAME      :")
    "+Nama)
        print("AGE      :")
    "+age)
        print("GENDER    :")
    "+gend)
        print("NRIC      :")
    "+Ic)
        print("EMAIL      :")
    "+Mail)
        print("PHONE NUMBER :")
    "+PhoneNum)
        print("WEIGHT     :")
    "+Weight)
        print("CATEGORY   :")
    "+category)
        ktgoty["text"]=category
            #set text in ktgoty to value of category
        print()
            #print black space
    except:
        #if error , do
        Weight="Invalid"
            #set Weight to invalid
        print("Weight
    "+Weight)
    Weight invalid
        OutputFrm["text"]=""
            #set text in OutputFrm to ""
        ktgotyLbl["text"]=""
            #set text in ktgotyLbl to ""
        ktgotyDot["text"]=""
            #set text in ktgotyDot to ""
        ktgoty["text"]=""
            #set text in ktgoty to ""
        OutputFrm.config(borderwidth=0)
            #set border width for OutputFrm to 0

    def
cncl():
    #cncl function
    try:

```

```

        #try to do
        projectdatabase =
mysql.connector.connect(                               #connect to
database
                                host="localhost",
                                user="root",
                                password="",
                                database="smars")

        mydbse =
projectdatabase.cursor()                           #Insta
ntiating a cursor for myprojectdb
        mydbse.execute("DELETE FROM muaythai WHERE
nric=%s",                                     #Executing database query with
parameters
                                (Ic,))
        projectdatabase.commit()
        print(f"Your Registration Canceled")

    except mysql.connector.Error as
err:                                              #if error , do
        print("Failed delete record :
{})".format(err)+"\n")                         #print error text

        messagebox.showinfo("Canceled","You canceled your
Registration")                      #show message box with text
        confwnds.withdraw()                    #hide the confwnds
        wnds.withdraw()                      #hide the wnds

def
comfrm():
    #comfrm function
    def
printrest():
    #function printrest
        print("\n\n          ***** Receipt
*****\n"+                                #print text
          "***** Sports Martial Arts : "+SMA+" *****\n")
        print("\tNAME      :")                  #print output
"+Nama)
        print("\tAGE      :")                  #print output
"+age)
        print("\tGENDER   :")                  #print output
"+gend)
        print("\tNRIC     :")                  #print output
"+Ic)
        print("\tEMAIL    :")                  #print output
"+Mail)
        print("\tPHONE NUMBER :")             #print output
"+PhoneNum)
        print("\tWEIGHT   :")                  #print output

```



```

aDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disable')
    #make a Entrytly disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER")
    #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disable')
    #make a Entrytly disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC")
    #Create Label for NRIC
nLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
nDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disable')
    #make n Entrytly disable
n.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-
MAIL")
    #Create Label for E-MAIL
eLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
eDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding

```

```

e=prjk.Entry(DataFrame)
    #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disable')
    #make e Entrytly disable
e.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE
NUMBER")
    #Create Label for PHONE NUMBER
pLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
pDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disable')
    #make p Entrytly disable
p.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT")
    #Create Label for WEIGHT
wLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
wDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry
w.insert(prjk.END,Weight)
    #insert Weight to w Entry
w.config(state='disable')
    #make w Entrytly disable
w.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY")
    #Create Label for CATEGORY
cLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
cDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry

```

```

        c.insert(prjk.END,category)
            #insert category to c Entry
        c.config(state='disabled')
            #make c Entry disable
        c.grid(row=6,column=2,padx=5,pady=5)
            #set position with grid and padding

        rstdwnds.mainloop()
            #Execute rstdwnds event loop
        print("You Successfully Registered in
"+SMA+"\n")                                #print text
        messagebox.showinfo("Successfully","Your data inserted
successfully")                            #show message box with text
        confwnds.withdraw()
            #hide the confwnds
        wndws.withdraw()
            #hide the wndws
        printrest()
            #call printrest function
        resit()
            #call resit function

        print("Please Confirm Your
Data\n")                                     #print text
        confwnds =
prjk.Tk()
#Create Tkinter window
        confwnds.geometry("375x450")
            #Set window size = 400 X 650
        confwnds.title("Confirm Your
Data")                                         #Set title
        confwnds.configure(bg="#faf8f5")
            #Set Background color

        confrmlbl=prjk.Label(confwnds,text="\n *** Please Confirm Your
Data ***\n"+      #Create Label
            " Sports Martial Arts : Muay Thai\n",
            bg="#faf8f5",font=("Times New Roman",18))
        confrmlbl.grid(row=0,column=0)
            #set position with grid and padding

        DataFrame=prjk.LabelFrame(confwnds,text="Your Information :
",
            #Create frame with title
            padx=10,pady=10,bg="#faf8f5")
        DataFrame.grid(row=1,column=0,padx=10)
            #set position with grid and padding

        nLbl=prjk.Label(DataFrame,text="NAME",
bg="#faf8f5")                                #Create Label for NAME
        nLbl.grid(row=0,column=0,padx=5,pady=5)
            #set position with grid and padding
        nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :

```

```

nDot.grid(row=0,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,Nama)
    #insert Nama to n Entry
n.config(state='disabled')
    #make n Entrytly disable
n.grid(row=0,column=2,padx=5,pady=5)
    #set position with grid and padding

aLbl=prjk.Label(DataFrame,text="AGE",
bg="#faf8f5")                      #Create Label for AGE
aLbl.grid(row=1,column=0,padx=5,pady=5)
    #set position with grid and padding
aDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
aDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disabled')
    #make a Entrtly disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER",
bg="#faf8f5")                      #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disabled')
    #make a Entrtly disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC",
bg="#faf8f5")                      #Create Label for  NRIC
nLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
nDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding

```

```

n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disabled')
    #make n Entrytly disable
n.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-MAIL",
bg="#faf8f5")                      #Create Label for E-MAIL
eLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
eDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
e=prjk.Entry(DataFrame)
    #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disabled')
    #make e Entrtly disable
e.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE NUMBER",
bg="#faf8f5")                      #Create Label for PHONE NUMBER
pLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
pDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disabled')
    #make p Entrtly disable
p.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT",
bg="#faf8f5")                      #Create Label for WEIGHT
wLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
wDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry

```

```

w.insert(prjk.END,Weight)
    #insert Weight to w Entry
w.config(state='disabled')
    #make w Entry disable
w.grid(row=6,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY",
bg="#faf8f5")                         #Create Label for CATEGORY
cLbl.grid(row=7,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                          #Create Label for :
cDot.grid(row=7,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry
c.insert(prjk.END,category)
    #insert category to c Entry
c.config(state='disabled')
    #make c Entry disable
c.grid(row=7,column=2,padx=5,pady=5)
    #set position with grid and padding

FrameBtn=prjk.Frame(confwnds,padx=10,bg="#faf8f5")
    #Create frame
FrameBtn.grid(row=2,column=0,padx=10,pady=10)
    #set position with grid and padding

BtnSub=prjk.Button(FrameBtn,text="CONFIRM",command=comfrm)
    #Create Confirm button
BtnSub.grid(row=0,column=0,padx=5)
    #set position with grid and padding

BtnCncl=prjk.Button(FrameBtn,text="CANCEL",command=cncl)
    #Create Cancel button
BtnCncl.grid(row=0,column=1,padx=5)
    #set position with grid and padding

SMA="Muay
Thai"
#Set SMA = Muay Thai
try:
    #try to do
    projectdatabase =
mysql.connector.connect(                               #connect to
database
    host="localhost",
    user="root",
    password="",
    database="smars")

mydbse =

```



```

        FillFrm=prjk.LabelFrame(wnds, text="Fill Your Information Here :
",
                           #Create frame with title
                           padx=10, pady=10, bg="#faf8f5")
FillFrm.grid(row=1,column=0,padx=10)
                           #set position with grid and padding

        nameLbl=prjk.Label(FillFrm,text="NAME",
bg="#faf8f5")                                     #Create Label for NAME
nameLbl.grid(row=0,column=0,padx=5,pady=5)
                           #set position with grid and padding
        nameDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
nameDot.grid(row=0,column=1,padx=5,pady=5)
                           #set position with grid and padding
        name=prjk.Entry(FillFrm)
                           #Create name Entry
name.grid(row=0,column=2,padx=5,pady=5)
                           #set position with grid and padding

        NrIcLbl=prjk.Label(FillFrm,text="NRIC",
bg="#faf8f5")                                     #Create Label for NRIC
NrIcLbl.grid(row=1,column=0,padx=5,pady=5)
                           #set position with grid and padding
        NrIcDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
NrIcDot.grid(row=1,column=1,padx=5,pady=5)
                           #set position with grid and padding
        NrIc=prjk.Entry(FillFrm)
                           #Create NrIc Entry
NrIc.grid(row=1,column=2,padx=5,pady=5)
                           #set position with grid and padding
        NrIcex=prjk.Label(FillFrm,text="Example :
010203040506",                                         #Create Label for Example
bg="#faf8f5",font=("Times New Roman",9))
NrIcex.grid(row=1,column=3,padx=5,pady=5)
                           #set position with grid and padding

        EMailLbl=prjk.Label(FillFrm,text="E-MAIL",
bg="#faf8f5")                                     #Create Label for E-MAIL
EMailLbl.grid(row=2,column=0,padx=5,pady=5)
                           #set position with grid and padding
        EMailDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
EMailDot.grid(row=2,column=1,padx=5,pady=5)
                           #set position with grid and padding
        EMail=prjk.Entry(FillFrm)
                           #Create EMail Entry
EMail.grid(row=2,column=2,padx=5,pady=5)
                           #set position with grid and padding

        PhoneLbl=prjk.Label(FillFrm,text="PHONE NUMBER",
bg="#faf8f5")                                     #Create Label for PHONE NUMBER
PhoneLbl.grid(row=3,column=0,padx=5,pady=5)

```

```

        #set position with grid and padding
PhoneDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
PhoneDot.grid(row=3,column=1,padx=5,pady=5)
        #set position with grid and padding
Phone=prjk.Entry(FillFrm)
        #Create Phone Entry
Phone.grid(row=3,column=2,padx=5,pady=5)
        #set position with grid and padding

BrtLbl=prjk.Label(FillFrm,text="WEIGHT",
bg="#faf8f5")                                     #Create Label for WEIGHT
BrtLbl.grid(row=4,column=0,padx=5,pady=5)
        #set position with grid and padding
BrtDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
BrtDot.grid(row=4,column=1,padx=5,pady=5)
        #set position with grid and padding
Brt=prjk.Entry(FillFrm)
        #Create Brt Entry
Brt.grid(row=4,column=2,padx=5,pady=5)
        #set position with grid and padding

BtnFrm=prjk.Frame(wnds,padx=10,bg="#faf8f5")
        #Create frame
BtnFrm.grid(row=5,column=0,padx=10,pady=10)
        #set position with grid and padding

submitBtn=prjk.Button(BtnFrm,text="SUBMIT",command=mytosbmt)
        #Create Submit button
submitBtn.grid(row=0,column=0,padx=5)
        #set position with grid and padding

rstBtn=prjk.Button(BtnFrm,text="RESET",command=mytorst)
        #Create Reset button
rstBtn.grid(row=0,column=1,padx=5)
        #set position with grid and padding

OutputFrm=prjk.LabelFrame(wnds,
        #Create frame with title
text="",borderwidth=0,
padx=10,pady=10,bg="#faf8f5")
OutputFrm.grid(row=6,column=0,padx=10)
        #set position with grid and padding

ktgotyLbl=prjk.Label(OutputFrm,text="",
        #Create Label
bg="#faf8f5")
ktgotyLbl.grid(row=0,column=0,padx=5,pady=5)
        #set position with grid and padding
ktgotyDot=prjk.Label(OutputFrm,text="",
        #Create Label
bg="#faf8f5")

```

```
ktgotyDot.grid(row=0,column=1,padx=5,pady=5)
    #set position with grid and padding
ktgoty=prjk.Label(OutputFrm,text="",
    #Create Label
bg="#faf8f5")
ktgoty.grid(row=0,column=2,padx=5,pady=5)
    #set position with grid and padding

wnds.mainloop()
#Exercute rsts wnds event loop
```

```
SMARS/ SILAT.py:
import tkinter as
prjk
#Use tkinter library for Python GUI
from Kategory import ktgory as
ctgory
in ktgory.py from Kategory file
from tkinter import
messagebox
import messagebox from tkinter library
import
mysql.connector
    #Imports MySQL Connector

def
silat():
    #silat function
    def
sltrst():
    #function sltrst
    name.delete(0,prjk.END)
        #clear input in name Entry
    NrIc.delete(0,prjk.END)
        #clear input in NrIc Entry
    EMail.delete(0,prjk.END)
        #clear input in EMail Entry
    Phone.delete(0,prjk.END)
        #clear input in Phone Entry
    Brt.delete(0,prjk.END)
        #clear input in Brt Entry
    OutputFrm["text"]=""
        #set text in OutputFrm to ""
    ktgotyLbl["text"]=""
        #set text in ktgotyLbl to ""
    ktgotyDot["text"]=""
        #set text in ktgotyDot to ""
    ktgoty["text"]=""
        #set text in ktgoty to ""
    OutputFrm.config(borderwidth=0)
        #set border width to OutputFrm to 0

    def
sltsbmt():
    #function sltsbmt
    OutputFrm["text"]="Your Category :
"
        #set text in OutputFrm
    OutputFrm.config(borderwidth=1)
        #set border width to OutputFrm to 1
    ktgotyLbl["text"]="Category"
        #set text in ktgotyLbl
    ktgotyDot["text"]":"
        #set text in ktgotyDot
```

```

Nama=name.get()
    #get name from value in name Entry
Ic=NrIc.get()
    #get Ic from value in NrIc Entry
Mail=EMail.get()
    #get Mail from value in EMail Entry
PhoneNum=Phone.get()
    #get PhoneNum from value in Phone Entry
Weight=float(Brt.get())
    #get float Weight from value in Brt Entry
try:
    #try to do
Nama=Nama.upper()
    #change name to upper
year=int(Ic[0:2])
    #get year from Ic parameter 0-2
if year>=0 and
year<=22:                                         #if year
0-22, do
    year+=2000
    #set year equal to year+2000
else:
    #id year >22 , do
    year+=1900
    #set year equal to year+1900
age=2023-
year
et age = 2023 - year                                         #s

jntina=int(Ic)
    #get jntina from Ic
jntina%=2
    #calculate jntina = jntina % 2
if jntina == 0:                                              #if
jntina equal to 0, do
    gend="FEMALE"
    #set gend to FEMALE
else:
    #IF jntina not equal to 0, do
    gend="MALE"
    #set gend to MALE

if
Weight>=1000:
    #if Weight > 1000, do
    Weight/=1000
        #Set Weight equal to Weight/1000
else:
    #if Weight < 1000
    Weight=Weight
        #Set Weight = Weight

```

```

        category=ctgory.Slt(Weight)
            #Set category and call Slt function from ctgory
        Weight="{:.2f}".format(Weight)
            #Set Weight to 2 decimal point
        Weight=Weight+
    kg"
    Weight to Weight+kg
        Ic=str(Ic)
            #Set Ic to string Ic
        age=str(age)
            #Set age to string age
        print("NAME      :")
    "+Nama)
        print("AGE      :")
    "+age)
        print("GENDER   :")
    "+gend)
        print("NRIC     :")
    "+Ic)
        print("EMAIL    :")
    "+Mail)
        print("PHONE NUMBER :")
    "+PhoneNum)
        print("WEIGHT   :")
    "+Weight)
        print("CATEGORY :")
    "+category)
        ktgoty["text"]=category
            #set text in ktgoty to value of category
        print()
            #print black space
    except:
        #if error , do
        Weight="Invalid"
            #set Weight to invalid
        print("Weight
    "+Weight)
    Weight invalid
        OutputFrm["text"]=""
            #set text in OutputFrm to ""
        ktgotyLbl["text"]=""
            #set text in ktgotyLbl to ""
        ktgotyDot["text"]=""
            #set text in ktgotyDot to ""
        ktgoty["text"]=""
            #set text in ktgoty to ""
        OutputFrm.config(borderwidth=0)
            #set border width for OutputFrm to 0

    def
cncl():
    #cncl function
    try:

```

```

        #try to do
        projectdatabase =
mysql.connector.connect(                               #connect to
database
                                host="localhost",
                                user="root",
                                password="",
                                database="smars")

        mydbse =
projectdatabase.cursor()                           #Insta
ntiating a cursor for myprojectdb
        mydbse.execute("DELETE FROM silat WHERE
nric=%s",                                         #Executing database query with
parameters
                                (Ic,))
        projectdatabase.commit()
        print(f"Your Registration Canceled")

    except mysql.connector.Error as
err:                                              #if error , do
        print("Failed delete record :
{})".format(err)+"\n")                         #print error text

        messagebox.showinfo("Canceled","You canceled your
Registration")                      #show message box with text
        confwnds.withdraw()                    #hide the confwnds
        wnds.withdraw()                      #hide the wnds

def
comfrm():
    #comfrm function
    def
printrest():
    #function printrest
        print("\n\n          ***** Receipt
*****\n"+                                #print text
          "***** Sports Martial Arts : "+SMA+" *****\n")
        print("\tNAME      :")                  #print output
"+Nama)
        print("\tAGE      :")                  #print output
"+age)
        print("\tGENDER   :")                  #print output
"+gend)
        print("\tNRIC     :")                  #print output
"+Ic)
        print("\tEMAIL    :")                  #print output
"+Mail)
        print("\tPHONE NUMBER :")             #print output
"+PhoneNum)
        print("\tWEIGHT   :")                  #print output

```



```

aDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disable')
    #make a Entrytly disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER")
    #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disable')
    #make a Entrytly disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC")
    #Create Label for NRIC
nLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
nDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disable')
    #make n Entrytly disable
n.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-
MAIL")
    #Create Label for E-MAIL
eLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" :")
    #Create Label for :
eDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding

```

```

e=prjk.Entry(DataFrame)
    #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disable')
    #make e Entrytly disable
e.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE
NUMBER")
    #Create Label for PHONE NUMBER
pLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
pDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disable')
    #make p Entrytly disable
p.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT")
    #Create Label for WEIGHT
wLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
wDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry
w.insert(prjk.END,Weight)
    #insert Weight to w Entry
w.config(state='disable')
    #make w Entrytly disable
w.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY")
    #Create Label for CATEGORY
cLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
cDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry

```

```

        c.insert(prjk.END,category)
            #insert category to c Entry
        c.config(state='disabled')
            #make c Entry disable
        c.grid(row=6,column=2,padx=5,pady=5)
            #set position with grid and padding

        rswndws.mainloop()
            #Execute rswndws event loop
        print("You Successfully Registered in
"+SMA+"\n")                                #print text
        messagebox.showinfo("Successfully","Your data inserted
successfully")                            #show message box with text
        confwnds.withdraw()
            #hide the confwnds
        wnds.withdraw()
            #hide the wnds
        printrest()
            #call printrest function
        resit()
            #call resit function

        print("Please Confirm Your
Data\n")                                     #print text
        confwnds =
prjk.Tk()
#Create Tkinter window
        confwnds.geometry("375x450")
            #Set window size = 400 X 650
        confwnds.title("Confirm Your
Data")                                         #Set title
        confwnds.configure(bg="#faf8f5")
            #Set Background color

        confrmlbl=prjk.Label(confwnds,text="\n *** Please Confirm Your
Data ***\n"+      #Create Label
            " Sports Martial Arts : Silat\n",
            bg="#faf8f5",font=("Times New Roman",18))
        confrmlbl.grid(row=0,column=0)
            #set position with grid and padding

        DataFrame=prjk.LabelFrame(confwnds,text="Your Information :
",
            #Create frame with title
            padx=10,pady=10,bg="#faf8f5")
        DataFrame.grid(row=1,column=0,padx=10)
            #set position with grid and padding

        nLbl=prjk.Label(DataFrame,text="NAME",
bg="#faf8f5")                                #Create Label for NAME
        nLbl.grid(row=0,column=0,padx=5,pady=5)
            #set position with grid and padding
        nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :

```

```

nDot.grid(row=0,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,Nama)
    #insert Nama to n Entry
n.config(state='disabled')
    #make n Entrytly disable
n.grid(row=0,column=2,padx=5,pady=5)
    #set position with grid and padding

aLbl=prjk.Label(DataFrame,text="AGE",
bg="#faf8f5")                                #Create Label for AGE
aLbl.grid(row=1,column=0,padx=5,pady=5)
    #set position with grid and padding
aDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
aDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disabled')
    #make a Entrtly disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER",
bg="#faf8f5")                                #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disabled')
    #make a Entrtly disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC",
bg="#faf8f5")                                #Create Label for  NRIC
nLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
nDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding

```

```

n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disabled')
    #make n Entrytly disable
n.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-MAIL",
bg="#faf8f5")                      #Create Label for E-MAIL
eLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
eDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
e=prjk.Entry(DataFrame)
    #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disabled')
    #make e Entrtly disable
e.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE NUMBER",
bg="#faf8f5")                      #Create Label for PHONE NUMBER
pLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
pDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disabled')
    #make p Entrtly disable
p.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT",
bg="#faf8f5")                      #Create Label for WEIGHT
wLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
wDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry

```

```

w.insert(prjk.END,Weight)
    #insert Weight to w Entry
w.config(state='disabled')
    #make w Entry disable
w.grid(row=6,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY",
bg="#faf8f5")                         #Create Label for CATEGORY
cLbl.grid(row=7,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                          #Create Label for :
cDot.grid(row=7,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry
c.insert(prjk.END,category)
    #insert category to c Entry
c.config(state='disabled')
    #make c Entry disable
c.grid(row=7,column=2,padx=5,pady=5)
    #set position with grid and padding

FrameBtn=prjk.Frame(confwnds,padx=10,bg="#faf8f5")
    #Create frame
FrameBtn.grid(row=2,column=0,padx=10,pady=10)
    #set position with grid and padding

BtnSub=prjk.Button(FrameBtn,text="CONFIRM",command=comfrm)
    #Create Confirm button
BtnSub.grid(row=0,column=0,padx=5)
    #set position with grid and padding

BtnCncl=prjk.Button(FrameBtn,text="CANCEL",command=cncl)
    #Create Cancel button
BtnCncl.grid(row=0,column=1,padx=5)
    #set position with grid and padding

SMA="Silat"
    #Set SMA = Muay Thai
try:
    #try to do
    projectdatabase =
mysql.connector.connect()                      #connect to
database
    host="localhost",
    user="root",
    password="",
    database="smars")

mydbse =
projectdatabase.cursor()                      #I

```

```

instantiating a cursor for myprojectdb
    mydbse.execute("SELECT * FROM silat WHERE
nric=%s",
                                #Executing database query with
parameters
                                (Ic,))
    sameinpt=mydbse.fetchone()
                                #Fetching one row from database

        if
sameinpt:
    #if sameinpt, do
        print("You Already Registered in
"+SMA+"\n")                                #print text
        messagebox.showinfo("Registered","You Already Registered
in "
                                #show message box with text
                                +SMA)
        confwnds.withdraw()
                                #hide the confwnds
        wnds.withdraw()
                                #hide the wnds
    else:
        #if not sameinpt, do
        mydbse.execute("INSERT INTO
silat"                                #insert data into
customers table
                                "(nama,age,nric,gender,email,phonenum,weight,category
)")
        "VALUES(%s,%s,%s,%s,%s,%s,%s,%s)",
        (Nama,age,Ic,gend,Mail,PhoneNum,Weight,category))
        projectdatabase.commit()
                                #Save changes to database

    except mysql.connector.Error as
err:                                #if error , do
        print("Failed Insert data :
{}".format(err)+"\n")                                #print error text

    wnds=prjk.Tk()
                                #Create Tkinter window
    wnds.geometry("444x420")
                                #Set winow size = 400 X 650
    wnds.title("Silat Registration
System")                                #Set title
    wnds.configure(bg="#faf8f5")
                                #Set Background color

    wlcm=prjk.Label(wnds,text="\n ***Welcome To Silat Registration
System***\n",
                                #Create Label
                                bg="#faf8f5",font=("Times New Roman",14))
    wlcm.grid(row=0,column=0)
                                #set position with grid and padding

    FillFrm=prjk.LabelFrame(wnds, text="Fill Your Information Here :

```

```

        #Create frame with title
        padx=10, pady=10, bg="#faf8f5")
FillFrm.grid(row=1,column=0,padx=10)
    #set position with grid and padding

    nameLbl=prjk.Label(FillFrm,text="NAME",
bg="#faf8f5")                                #Create Label for NAME
    nameLbl.grid(row=0,column=0,padx=5,pady=5)
        #set position with grid and padding
    nameDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                #Create Label for :
    nameDot.grid(row=0,column=1,padx=5,pady=5)
        #set position with grid and padding
    name=prjk.Entry(FillFrm)
        #Create name Entry
    name.grid(row=0,column=2,padx=5,pady=5)
        #set position with grid and padding

    NrIcLbl=prjk.Label(FillFrm,text="NRIC",
bg="#faf8f5")                                #Create Label for NRIC
    NrIcLbl.grid(row=1,column=0,padx=5,pady=5)
        #set position with grid and padding
    NrIcDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                #Create Label for :
    NrIcDot.grid(row=1,column=1,padx=5,pady=5)
        #set position with grid and padding
    NrIc=prjk.Entry(FillFrm)
        #Create NrIc Entry
    NrIc.grid(row=1,column=2,padx=5,pady=5)
        #set position with grid and padding
    NrIcex=prjk.Label(FillFrm,text="Example :
010203040506",                               #Create Label for Example
bg="#faf8f5",font=("Times New Roman",9))
    NrIcex.grid(row=1,column=3,padx=5,pady=5)
        #set position with grid and padding

    EMailLbl=prjk.Label(FillFrm,text="E-MAIL",
bg="#faf8f5")                                #Create Label for E-MAIL
    EMailLbl.grid(row=2,column=0,padx=5,pady=5)
        #set position with grid and padding
    EMailDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                #Create Label for :
    EMailDot.grid(row=2,column=1,padx=5,pady=5)
        #set position with grid and padding
    EMail=prjk.Entry(FillFrm)
        #Create EMail Entry
    EMail.grid(row=2,column=2,padx=5,pady=5)
        #set position with grid and padding

    PhoneLbl=prjk.Label(FillFrm,text="PHONE NUMBER",
bg="#faf8f5")                                #Create Label for PHONE NUMBER
    PhoneLbl.grid(row=3,column=0,padx=5,pady=5)
        #set position with grid and padding

```

```

PhoneDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
PhoneDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding
Phone=prjk.Entry(FillFrm)
    #Create Phone Entry
Phone.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

BrtLbl=prjk.Label(FillFrm,text="WEIGHT",
bg="#faf8f5")                                     #Create Label for WEIGHT
BrtLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
BrtDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
BrtDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
Brt=prjk.Entry(FillFrm)
    #Create Brt Entry
Brt.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

BtnFrm=prjk.Frame(wnds,padx=10,bg="#faf8f5")
    #Create frame
BtnFrm.grid(row=5,column=0,padx=10,pady=10)
    #set position with grid and padding

submitBtn=prjk.Button(BtnFrm,text="SUBMIT",command=s1tsbmt)
    #Create Submit button
submitBtn.grid(row=0,column=0,padx=5)
    #set position with grid and padding

rstBtn=prjk.Button(BtnFrm,text="RESET",command=s1trst)
    #Create Reset button
rstBtn.grid(row=0,column=1,padx=5)
    #set position with grid and padding

OutputFrm=prjk.LabelFrame(wnds,
    #Create frame with title
text="",borderwidth=0,
padx=10,pady=10,bg="#faf8f5")
OutputFrm.grid(row=6,column=0,padx=10)
    #set position with grid and padding

ktgotyLbl=prjk.Label(OutputFrm,text="",
    #Create Label
bg="#faf8f5")
ktgotyLbl.grid(row=0,column=0,padx=5,pady=5)
    #set position with grid and padding
ktgotyDot=prjk.Label(OutputFrm,text="",
    #Create Label
bg="#faf8f5")
ktgotyDot.grid(row=0,column=1,padx=5,pady=5)

```

```
        #set position with grid and padding
ktgoty=prjk.Label(OutputFrm,text="",
                  #Create Label
bg="#faf8f5")
ktgoty.grid(row=0,column=2,padx=5,pady=5)
                  #set position with grid and padding

wnds.mainloop()
                  #Exercute rstmnds event loop
```

```

SMARS/ TAEKWONDO.py:
import tkinter as
prjk
#Use tkinter library for Python GUI
from Kategory import ktgory as
ctgory
in ktgory.py from Kategory file
from tkinter import
messagebox
import messagebox from tkinter library
import
mysql.connector
    #Imports MySQL Connector

def
taekwondo():
    #taekwondo function
    def
tnedorst():
    #function tndorst
    name.delete(0,prjk.END)
        #clear input in name Entry
    NrIc.delete(0,prjk.END)
        #clear input in NrIc Entry
    EMail.delete(0,prjk.END)
        #clear input in EMail Entry
    Phone.delete(0,prjk.END)
        #clear input in Phone Entry
    Brt.delete(0,prjk.END)
        #clear input in Brt Entry
    OutputFrm["text"]=""
        #set text in OutputFrm to ""
    ktgotyLbl["text"]=""
        #set text in ktgotyLbl to ""
    ktgotyDot["text"]=""
        #set text in ktgotyDot to ""
    ktgoty["text"]=""
        #set text in ktgoty to ""
    OutputFrm.config(borderwidth=0)
        #set border width to OutputFrm to 0

    def
tndosbmt():
    #function tndosbmt
    OutputFrm["text"]="Your Category :
"
        #set text in OutputFrm
    OutputFrm.config(borderwidth=1)
        #set border width to OutputFrm to 1
    ktgotyLbl["text"]="Category"
        #set text in ktgotyLbl
    ktgotyDot["text"]":"
        #set text in ktgotyDot

```

```

Nama=name.get()
    #get name from value in name Entry
Ic=NrIc.get()
    #get Ic from value in NrIc Entry
Mail=EMail.get()
    #get Mail from value in EMail Entry
PhoneNum=Phone.get()
    #get PhoneNum from value in Phone Entry
Weight=float(Brt.get())
    #get float Weight from value in Brt Entry
try:
    #try to do
Nama=Nama.upper()
    #change name to upper
year=int(Ic[0:2])
    #get year from Ic parameter 0-1
if year>=0 and
year<=22:                                         #if year
0-22, do
    year+=2000
    #set year equal to year+2000
else:
    #id year >22 , do
    year+=1900
    #set year equal to year+1900
age=2023-
year
et age = 2023 - year                                         #s

jntina=int(Ic)
    #get jntina from Ic
jntina%=2
    #calculate jntina = jntina % 2
if jntina == 0:                                              #if
jntina equal to 0, do
    gend="FEMALE"
    #set gend to FEMALE
else:
    #IF jntina not equal to 0, do
    gend="MALE"
    #set gend to MALE

if
Weight>=1000:
    #if Weight > 1000, do
    Weight/=1000
        #Set Weight equal to Weight/1000
else:
    #if Weight < 1000
    Weight=Weight
        #Set Weight = Weight

```

```

        category=ctgory.Twdo(Weight)
            #Set category and call Twdo function from ctgory
        Weight="{:.2f}".format(Weight)
            #Set Weight to 2 decimal point
        Weight=Weight+
    kg"
    Weight to Weight+kg
        Ic=str(Ic)
            #Set Ic to string Ic
        age=str(age)
            #Set age to string age
        print("NAME      :")
    "+Nama)
        print("AGE      :")
    "+age)
        print("GENDER   :")
    "+gend)
        print("NRIC     :")
    "+Ic)
        print("EMAIL    :")
    "+Mail)
        print("PHONE NUMBER :")
    "+PhoneNum)
        print("WEIGHT   :")
    "+Weight)
        print("CATEGORY :")
    "+category)
        ktgoty["text"]=category
            #set text in ktgoty to value of category
        print()
            #print black space
    except:
        #if error , do
        Weight="Invalid"
            #set Weight to invalid
        print("Weight
    "+Weight)
    Weight invalid
        OutputFrm["text"]=""
            #set text in OutputFrm to ""
        ktgotyLbl["text"]=""
            #set text in ktgotyLbl to ""
        ktgotyDot["text"]=""
            #set text in ktgotyDot to ""
        ktgoty["text"]=""
            #set text in ktgoty to ""
        OutputFrm.config(borderwidth=0)
            #set border width for OutputFrm to 0

    def
cncl():
    #cncl function
    try:

```

```

        #try to do
        projectdatabase =
mysql.connector.connect(                               #connect to
database
                                host="localhost",
                                user="root",
                                password="",
                                database="smars")

        mydbse =
projectdatabase.cursor()                           #Insta
ntiating a cursor for myprojectdb
        mydbse.execute("DELETE FROM taekwondo WHERE
nric=%s",                                         #Executing database query with parameters
                                (Ic,))
        projectdatabase.commit()
        print(f"Your Registration Canceled")

    except mysql.connector.Error as
err:                                              #if error , do
        print("Failed delete record :
{}".format(err)+"\n")                            #print error text

        messagebox.showinfo("Canceled","You canceled your
Registration")                                #show message box with text
        confwnds.withdraw()                         #hide the confwnds
        wnds.withdraw()                            #hide the wnds

    def
comfrm():
    #comfrm function
    def
printrest():
    #function printrest
        print("\n\n          ***** Receipt
*****\n"+                                #print text
        "***** Sports Martial Arts : "+SMA+" *****\n")
        print("\tNAME      :")                      #print output
"+Nama)
        print("\tAGE      :")                      #print output
"+age)
        print("\tGENDER   :")                      #print output
"+gend)
        print("\tNRIC     :")                      #print output
"+Ic)
        print("\tEMAIL    :")                      #print output
"+Mail)
        print("\tPHONE NUMBER :")                  #print output
"+PhoneNum)
        print("\tWEIGHT   :")                      #print output
"+Weight)

```



```

        #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disable')
    #make a Entrytly disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER")
    #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" :
")                                     #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disable')
    #make a Entrty disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC")
    #Create Label for NRIC
nLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" :
")                                     #Create Label for :
nDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disable')
    #make n Entrty disable
n.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-
MAIL")                                #Create Label for E-MAIL
eLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" :
")                                     #Create Label for :
eDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding
e=prjk.Entry(DataFrame)

```

```

        #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disable')
    #make e Entrytly disable
e.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE
NUMBER")
    #Create Label for PHONE NUMBER
pLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
pDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disable')
    #make p Entrty disable
p.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT")
    #Create Label for WEIGHT
wLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
wDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry
w.insert(prjk.END,Weight)
    #insert Weight to w Entry
w.config(state='disable')
    #make w Entrty disable
w.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY")
    #Create Label for CATEGORY
cLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" :
")
    #Create Label for :
cDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry
c.insert(prjk.END,category)

```

```

        #insert category to c Entry
c.config(state='disabled')
    #make c Entry disable
c.grid(row=6,column=2,padx=5,pady=5)
    #set position with grid and padding

rstwnds.mainloop()
    #Execute rstwnds event loop
print("You Successfully Registered in
"+SMA+"\n")                                #print text
messagebox.showinfo("Successfully","Your data inserted
successfully")           #show message box with text
confwnds.withdraw()
    #hide the confwnds
wnds.withdraw()
    #hide the wnds
printrest()
    #call printrest function
resit()
    #call resit function

print("Please Confirm Your
Data\n")                                     #print text
confwnds =
prjk.Tk()
#Create Tkinter window
confwnds.geometry("375x450")
    #Set window size = 400 X 650
confwnds.title("Confirm Your
Data")                                         #Set title
confwnds.configure(bg="#faf8f5")
    #Set Background color

confrmlbl=prjk.Label(confwnds,text="\n *** Please Confirm Your
Data ***\n"+      #Create Label
    " Sports Martial Arts : Taekwondo\n",
    bg="#faf8f5",font=("Times New Roman",18))
confrmlbl.grid(row=0,column=0)
    #set position with grid and padding

DataFrame=prjk.LabelFrame(confwnds,text="Your Information :
",
    #Create frame with title
    padx=10,pady=10,bg="#faf8f5")
DataFrame.grid(row=1,column=0,padx=10)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NAME",
bg="#faf8f5")                                #Create Label for NAME
nLbl.grid(row=0,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
nDot.grid(row=0,column=1,padx=5,pady=5)

```

```

        #set position with grid and padding
n=prjk.Entry(DataFrame)
    #Create n Entry
n.insert(prjk.END,Nama)
    #insert Nama to n Entry
n.config(state='disabled')
    #make n Entry disable
n.grid(row=0,column=2,padx=5,pady=5)
    #set position with grid and padding

aLbl=prjk.Label(DataFrame,text="AGE",
bg="#faf8f5")                                #Create Label for AGE
aLbl.grid(row=1,column=0,padx=5,pady=5)
    #set position with grid and padding
aDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
aDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
a=prjk.Entry(DataFrame)
    #Create a Entry
a.insert(prjk.END,age)
    #insert age to a Entry
a.config(state='disabled')
    #make a Entry disable
a.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding

gLbl=prjk.Label(DataFrame,text="GENDER",
bg="#faf8f5")                                #Create Label for GENDER
gLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
gDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
gDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
g=prjk.Entry(DataFrame)
    #Create a Entry
g.insert(prjk.END,gend)
    #insert gend to a Entry
g.config(state='disabled')
    #make a Entry disable
g.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

nLbl=prjk.Label(DataFrame,text="NRIC",
bg="#faf8f5")                                #Create Label for NRIC
nLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
nDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
nDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding
n=prjk.Entry(DataFrame)

```

```

        #Create n Entry
n.insert(prjk.END,str(Ic))
    #insert string Ic to n Entry
n.config(state='disabled')
    #make n Entry disabled
n.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

eLbl=prjk.Label(DataFrame,text="E-MAIL",
bg="#faf8f5")                      #Create Label for E-MAIL
eLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
eDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
eDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
e=prjk.Entry(DataFrame)
    #Create e Entry
e.insert(prjk.END,Mail)
    #insert Mail to e Entry
e.config(state='disabled')
    #make e Entry disabled
e.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

pLbl=prjk.Label(DataFrame,text="PHONE NUMBER",
bg="#faf8f5")                      #Create Label for PHONE NUMBER
pLbl.grid(row=5,column=0,padx=5,pady=5)
    #set position with grid and padding
pDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
pDot.grid(row=5,column=1,padx=5,pady=5)
    #set position with grid and padding
p=prjk.Entry(DataFrame)
    #Create p Entry
p.insert(prjk.END,PhoneNum)
    #insert PhoneNum to p Entry
p.config(state='disabled')
    #make p Entry disabled
p.grid(row=5,column=2,padx=5,pady=5)
    #set position with grid and padding

wLbl=prjk.Label(DataFrame,text="WEIGHT",
bg="#faf8f5")                      #Create Label for WEIGHT
wLbl.grid(row=6,column=0,padx=5,pady=5)
    #set position with grid and padding
wDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                      #Create Label for :
wDot.grid(row=6,column=1,padx=5,pady=5)
    #set position with grid and padding
w=prjk.Entry(DataFrame)
    #Create w Entry
w.insert(prjk.END,Weight)

```

```

        #insert Weight to w Entry
w.config(state='disabled')
    #make w Entry disable
w.grid(row=6,column=2,padx=5,pady=5)
    #set position with grid and padding

cLbl=prjk.Label(DataFrame,text="CATEGORY",
bg="#faf8f5")                                #Create Label for CATEGORY
cLbl.grid(row=7,column=0,padx=5,pady=5)
    #set position with grid and padding
cDot=prjk.Label(DataFrame,text=" : ",
bg="#faf8f5")                                #Create Label for :
cDot.grid(row=7,column=1,padx=5,pady=5)
    #set position with grid and padding
c=prjk.Entry(DataFrame)
    #Create c Entry
c.insert(prjk.END,category)
    #insert category to c Entry
c.config(state='disabled')
    #make c Entry disable
c.grid(row=7,column=2,padx=5,pady=5)
    #set position with grid and padding

FrameBtn=prjk.Frame(confwnds,padx=10,bg="#faf8f5")
    #Create frame
FrameBtn.grid(row=2,column=0,padx=10,pady=10)
    #set position with grid and padding

BtnSub=prjk.Button(FrameBtn,text="CONFIRM",command=comfrm)
    #Create Confirm button
BtnSub.grid(row=0,column=0,padx=5)
    #set position with grid and padding

BtnCncl=prjk.Button(FrameBtn,text="CANCEL",command=cncl)
    #Create Cancel button
BtnCncl.grid(row=0,column=1,padx=5)
    #set position with grid and padding

SMA="Taekwondo"
    #Set SMA = Muay Thai
try:
    #try to do
    projectdatabase =
mysql.connector.connect(                               #connect to
database
    host="localhost",
    user="root",
    password="",
    database="smars")

mydbse =
projectdatabase.cursor()                           #I
instantiating a cursor for myprojectdb

```

```

        mydbse.execute("SELECT * FROM taekwondo WHERE
nric=%s",
                           #Executing database query with
parameters
                           (Ic,))
sameinpt=mydbse.fetchone()
#Fetching one row from database

if
sameinpt:
    #if sameinpt, do
        print("You Already Registered in
"+SMA+"\n")           #print text
        messagebox.showinfo("Registered","You Already Registered
in "
                           #show message box with text
                           +SMA)
        confwnds.withdraw()
                           #hide the confwnds
        wnds.withdraw()
                           #hide the wnds
    else:
        #if not sameinpt, do
        mydbse.execute("INSERT INTO
taekwondo"                      #insert data into
customers table
                           "(nama,age,nric,gender,email,phonenum,weight,category
)")
                           "VALUES(%s,%s,%s,%s,%s,%s,%s,%s)",
                           (Nama,age,Ic,gend,Mail,PhoneNum,Weight,category))
        projectdatabase.commit()
                           #Save changes to database

    except mysql.connector.Error as
err:                                #if error , do
        print("Failed Insert data :
{}".format(err)+"\n")                 #print error text

    wnds=prjk.Tk()
                           #Create Tkinter window
    wnds.geometry("444x420")
                           #Set winow size = 400 X 650
    wnds.title("Taekwondo Registration
System")                         #Set title
    wnds.configure(bg="#faf8f5")
                           #Set Background color

    wlcm=prjk.Label(wnds,text="\n ***Welcome To Taekwondo Registration
System***\n",
                     #Create Label
                     bg="#faf8f5",font=("Times New Roman",14))
    wlcm.grid(row=0,column=0)
                           #set position with grid and padding

    FillFrm=prjk.LabelFrame(wnds, text="Fill Your Information Here :
",
                           #Create frame with title

```

```

    padx=10, pady=10, bg="#faf8f5")
FillFrm.grid(row=1,column=0,padx=10)
    #set position with grid and padding

    nameLbl=prjk.Label(FillFrm,text="NAME",
bg="#faf8f5")                                     #Create Label for NAME
nameLbl.grid(row=0,column=0,padx=5,pady=5)
    #set position with grid and padding
    nameDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
nameDot.grid(row=0,column=1,padx=5,pady=5)
    #set position with grid and padding
    name=prjk.Entry(FillFrm)
        #Create name Entry
name.grid(row=0,column=2,padx=5,pady=5)
    #set position with grid and padding

    NrIcLbl=prjk.Label(FillFrm,text="NRIC",
bg="#faf8f5")                                     #Create Label for NRIC
NrIcLbl.grid(row=1,column=0,padx=5,pady=5)
    #set position with grid and padding
    NrIcDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
NrIcDot.grid(row=1,column=1,padx=5,pady=5)
    #set position with grid and padding
    NrIc=prjk.Entry(FillFrm)
        #Create NrIc Entry
NrIc.grid(row=1,column=2,padx=5,pady=5)
    #set position with grid and padding
    NrIcx=prjk.Label(FillFrm,text="Example :
010203040506",                                    #Create Label for Example
bg="#faf8f5",font=("Times New Roman",9))
NrIcx.grid(row=1,column=3,padx=5,pady=5)
    #set position with grid and padding

    EMailLbl=prjk.Label(FillFrm,text="E-MAIL",
bg="#faf8f5")                                     #Create Label for E-MAIL
EMailLbl.grid(row=2,column=0,padx=5,pady=5)
    #set position with grid and padding
    EMailDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
EMailDot.grid(row=2,column=1,padx=5,pady=5)
    #set position with grid and padding
    EMail=prjk.Entry(FillFrm)
        #Create EMail Entry
EMail.grid(row=2,column=2,padx=5,pady=5)
    #set position with grid and padding

    PhoneLbl=prjk.Label(FillFrm,text="PHONE NUMBER",
bg="#faf8f5")                                     #Create Label for PHONE NUMBER
PhoneLbl.grid(row=3,column=0,padx=5,pady=5)
    #set position with grid and padding
    PhoneDot=prjk.Label(FillFrm,text=" : ",

```

```

bg="#faf8f5")                                     #Create Label for :
PhoneDot.grid(row=3,column=1,padx=5,pady=5)
    #set position with grid and padding
Phone=prjk.Entry(FillFrm)
    #Create Phone Entry
Phone.grid(row=3,column=2,padx=5,pady=5)
    #set position with grid and padding

BrtLbl=prjk.Label(FillFrm,text="WEIGHT",
bg="#faf8f5")                                     #Create Label for WEIGHT
BrtLbl.grid(row=4,column=0,padx=5,pady=5)
    #set position with grid and padding
BrtDot=prjk.Label(FillFrm,text=" : ",
bg="#faf8f5")                                     #Create Label for :
BrtDot.grid(row=4,column=1,padx=5,pady=5)
    #set position with grid and padding
Brt=prjk.Entry(FillFrm)
    #Create Brt Entry
Brt.grid(row=4,column=2,padx=5,pady=5)
    #set position with grid and padding

BtnFrm=prjk.Frame(wnds,padx=10,bg="#faf8f5")
    #Create frame
BtnFrm.grid(row=5,column=0,padx=10,pady=10)
    #set position with grid and padding

submitBtn=prjk.Button(BtnFrm,text="SUBMIT",command=tndosbmt)
    #Create Submit button
submitBtn.grid(row=0,column=0,padx=5)
    #set position with grid and padding

rstBtn=prjk.Button(BtnFrm,text="RESET",command=tndorst)
    #Create Reset button
rstBtn.grid(row=0,column=1,padx=5)
    #set position with grid and padding

OutputFrm=prjk.LabelFrame(wnds,
    #Create frame with title
text="",borderwidth=0,
padx=10,pady=10,bg="#faf8f5")
OutputFrm.grid(row=6,column=0,padx=10)
    #set position with grid and padding

ktgotyLbl=prjk.Label(OutputFrm,text="",
    #Create Label
bg="#faf8f5")
ktgotyLbl.grid(row=0,column=0,padx=5,pady=5)
    #set position with grid and padding
ktgotyDot=prjk.Label(OutputFrm,text="",
    #Create Label
bg="#faf8f5")
ktgotyDot.grid(row=0,column=1,padx=5,pady=5)
    #set position with grid and padding

```

```
ktgoty=prjk.Label(OutputFrm,text="",
                   #Create Label
                   bg="#faf8f5")
ktgoty.grid(row=0,column=2,padx=5,pady=5)
                   #set position with grid and padding

wnds.mainloop()
                   #Exercute rsts wnds event loop
```

## INTERFACE OF OUR SYSTEM :

SportMartialArtsRegSystem.py:



SMARS/ BOXING.py:

BOXING clicked!

Boxing Registration System

\*\*\*Welcome To Boxing Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/>
NRIC	:	<input type="text"/> Example : 010203040506
E-MAIL	:	<input type="text"/>
PHONE NUMBER	:	<input type="text"/>
WEIGHT	:	<input type="text"/>

NAME : DANISH FADILAH  
AGE : 22  
GENDER : MALE  
NRIC : 010601060161  
EMAIL : dfadil@gmail.com  
PHONE NUMBER : 011-3310 1234  
WEIGHT : 47.90 kg  
CATEGORY : Welterweight (-66kg)

Boxing Registration System

\*\*\*Welcome To Boxing Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> DanISH FadiLAH
NRIC	:	<input type="text"/> 010601060161 Example : 010203040506
E-MAIL	:	<input type="text"/> dfadil@gmail.com
PHONE NUMBER	:	<input type="text"/> 011-3310 1234
WEIGHT	:	<input type="text"/> 47.9

Your Category :  
Category : Welterweight (-66kg)

Boxing Registration System

\*\*\*Welcome To Boxing Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> DanISH FadiLAH
NRIC	:	<input type="text"/> 010601060161 Example : 010203040506
E-MAIL	:	<input type="text"/> dfadil@gmail.com
PHONE NUMBER	:	<input type="text"/> 011-3310 1234
WEIGHT	:	<input type="text"/> 47.90 kg
CATEGORY	:	<input type="text"/> Welterweight (-66kg)

Successfully

Your data inserted successfully

OK

You Successfully Registered in Boxing

\*\*\*\*\* Receipt \*\*\*\*\*

\*\*\*\*\* Sports Martial Arts : Boxing \*\*\*\*\*

NAME : DANISH FADILAH  
AGE : 22  
GENDER : MALE  
NRIC : 010601060161  
EMAIL : dfadil@gmail.com  
PHONE NUMBER : 011-3310 1234  
WEIGHT : 47.90 kg  
CATEGORY : Welterweight (-66kg)

Your Registration Receipt

\*\*\* Receipt \*\*\*

\*\*\* Sports Martial Arts : Boxing \*\*\*

Your Information :

NAME	:	DANISH FADILAH
AGE	:	22
NRIC	:	010601060161
E-MAIL	:	dfadil@gmail.com
PHONE NUMBER	:	011-3310 1234
WEIGHT	:	47.90 kg
CATEGORY	:	Welterweight (-66kg)

nama	age	nric	gender	email	phonenum	weight	category
DANISH FADILAH	22	010601060161	MALE	dfadil@gmail.com	011-3310 1234	47.90 kg	Welterweight (-66kg)

SMARS/ KARATE.py:

KARATE clicked!

Karate Registration System

\*\*\*Welcome To Karate Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/>
NRIC	:	<input type="text"/> Example : 010203040506
E-MAIL	:	<input type="text"/>
PHONE NUMBER	:	<input type="text"/>
WEIGHT	:	<input type="text"/>

SUBMIT    RESET

NAME : 'AINUL HAFIZ' (highlighted in red)

AGE : 26

GENDER : MALE

NRIC : 970717077747

EMAIL : ainulhafz@gmail.com

PHONE NUMBER : 0175561707

WEIGHT : 86.97 kg

CATEGORY : +80kg

Karate Registration System

\*\*\*Welcome To Karate Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> 'AINUL HAFIZ
NRIC	:	<input type="text"/> 970717077747 Example : 010203040506
E-MAIL	:	<input type="text"/> ainulhafz@gmail.com
PHONE NUMBER	:	<input type="text"/> 0175561707
WEIGHT	:	<input type="text"/> 86974

SUBMIT    RESET

Your Category :

Category : +80kg

Karate Registration System

\*\*\*Welcome To Karate Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> 'AINUL HAFIZ
NRIC	:	<input type="text"/> 26
E-MAIL	:	<input type="text"/> 970717077747
PHONE NUMBER	:	<input type="text"/> ainulhafz@gmail.com
WEIGHT	:	<input type="text"/> 0175561707
CATEGORY	:	<input type="text"/> 86.97 kg

CONFIRM    CANCEL

Successfully

Your data inserted successfully

OK

You Successfully Registered in Karate

\*\*\*\*\* Receipt \*\*\*\*\*  
\*\*\*\*\* Sports Martial Arts : Karate \*\*\*\*\*  
NAME : 'AINUL HAFIZ  
AGE : 26  
GENDER : MALE  
NRIC : 970717077747  
EMAIL : ainulhafz@gmail.com  
PHONE NUMBER : 0175561707  
WEIGHT : 86.97 kg  
CATEGORY : +80kg  
\*\*\*\*\*

Your Registration Receipt

\*\*\* Receipt \*\*\*  
\*\*\* Sports Martial Arts : Karate \*\*\*

Your Information :

NAME	:	AINUL HAFIZ
AGE	:	26
GENDER	:	MALE
NRIC	:	970717077747
E-MAIL	:	ainulhafz@gmail.com
PHONE NUMBER	:	0175561707
WEIGHT	:	86.97 kg
CATEGORY	:	+80kg

nama	age	nric	gender	email	phonenum	weight	category
'AINUL HAFIZ	26	970717077747	MALE	ainulhafz@gmail.com	0175561707	86.97 kg	+80kg

SMARS/ MMA.py:

MMA clicked!

\*\*\*Welcome To MMA Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/>
NRIC	:	<input type="text"/> Example : 010203040506
E-MAIL	:	<input type="text"/>
PHONE NUMBER	:	<input type="text"/>
WEIGHT	:	<input type="text"/>

**SUBMIT** **RESET**

NAME : MOHAMAD ALIFF  
AGE : 34  
GENDER : MALE  
NRIC : 890720041041  
EMAIL : alip@gmail.com  
PHONE NUMBER : 0112421014  
WEIGHT : 123.69 kg  
CATEGORY : Super Heavyweight (+121kg)

\*\*\*Welcome To MMA Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> mOhAMad ALiFF
NRIC	:	<input type="text"/> 890720041041 Example : 010203040506
E-MAIL	:	<input type="text"/> alip@gmail.com
PHONE NUMBER	:	<input type="text"/> 0112421014
WEIGHT	:	<input type="text"/> 123689

**SUBMIT** **RESET**

Your Category :  
Category : Super Heavyweight (+121kg)

\*\*\*Welcome To MMA Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> mOhAMad ALIFF
NRIC	:	<input type="text"/> 890720041041 Example : 010203040506
E-MAIL	:	<input type="text"/> alip@gmail.com
PHONE NUMBER	:	<input type="text"/> 0112421014
WEIGHT	:	<input type="text"/> 123689

Please Confirm Your Data

\*\*\* Please Confirm Your Data \*\*\*  
Sports Martial Arts : MMA

Your Information :

NAME	:	<input type="text"/> MOHAMAD ALIFF
AGE	:	<input type="text"/> 34
GENDER	:	<input type="text"/> MALE
NRIC	:	<input type="text"/> 890720041041
E-MAIL	:	<input type="text"/> alip@gmail.com
PHONE NUMBER	:	<input type="text"/> 0112421014
WEIGHT	:	<input type="text"/> 123.69 kg
CATEGORY	:	<input type="text"/> Super Heavyweight (+121kg)

**CONFIRM** **CANCEL**

Successfully

Your data inserted successfully

**OK**

You Successfully Registered in MMA

\*\*\*\*\* Receipt \*\*\*\*\*  
\*\*\*\*\* Sports Martial Arts : MMA \*\*\*\*\*

NAME	:	MOHAMAD ALIFF
AGE	:	34
GENDER	:	MALE
NRIC	:	890720041041
EMAIL	:	alip@gmail.com
PHONE NUMBER	:	0112421014
WEIGHT	:	123.69 kg
CATEGORY	:	Super Heavyweight (+121kg)

\*\*\*\*\*

Your Registration Receipt

\*\*\* Receipt \*\*\*  
\*\*\* Sports Martial Arts : MMA \*\*\*

-Your Information :

NAME	:	MOHAMAD ALIFF
AGE	:	34
GENDER	:	MALE
NRIC	:	890720041041
E-MAIL	:	alip@gmail.com
PHONE NUMBER	:	0112421014
WEIGHT	:	123.69 kg
CATEGORY	:	Super Heavyweight (+1

nama	age	nruc	gender	email	phonenum	weight	category
MOHAMAD ALIFF	34	890720041041	MALE	alip@gmail.com	0112421014	123.69 kg	Super Heavyweight (+121kg)

SMARS/ MUAYTHAI .py:

MUAYTHAI clicked!

\*\*\*Welcome To Muay Thai Registration System\*\*\*

Fill Your Information Here :—

NAME :	<input type="text"/>
NRIC :	<input type="text"/> Example : 010203040506
E-MAIL :	<input type="text"/>
PHONE NUMBER :	<input type="text"/>
WEIGHT :	<input type="text"/>

**SUBMIT** **RESET**

MUAYTHAI clicked!

NAME : NUR HIDAYAH  
AGE : 17  
GENDER : FEMALE  
NRIC : 060309070104  
EMAIL : dayah@gmail.com  
PHONE NUMBER : 018-94 9876  
WEIGHT : 48.98 kg  
CATEGORY : Mini Flyweigh (-48kg)

\*\*\*Welcome To Muay Thai Registration System\*\*\*

Fill Your Information Here :—

NAME :	<input type="text"/> Nur HiDaYAH
NRIC :	<input type="text"/> 060309070104 Example : 010203040506
E-MAIL :	<input type="text"/> dayah@gmail.com
PHONE NUMBER :	<input type="text"/> 018-94 9876
WEIGHT :	<input type="text"/> 48977

**SUBMIT** **RESET**

Your Category :—  
Category : Mini Flyweigh (-48kg)

Muay Thai Registration System

\*\*\*Welcome To Muay Thai Registration System\*\*\*

Fill Your Information Here :—

NAME :	<input type="text"/> Nur HiDaYAH
NRIC :	<input type="text"/> 060309070104 Example : 010203040506
E-MAIL :	<input type="text"/> dayah@gmail.com
PHONE NUMBER :	<input type="text"/> 018-94 9876
WEIGHT :	<input type="text"/> 48977

**SUBMIT** **RESET**

Please Confirm Your Data

\*\*\* Please Confirm Your Data \*\*\*  
Sports Martial Arts : Muay Thai

Your Information :—

NAME :	<input type="text"/> NUR HIDAYAH
AGE :	<input type="text"/> 17
GENDER :	<input type="text"/> FEMALE
NRIC :	<input type="text"/> 060309070104
E-MAIL :	<input type="text"/> dayah@gmail.com
PHONE NUMBER :	<input type="text"/> 018-94 9876
WEIGHT :	<input type="text"/> 48.98 kg
CATEGORY :	<input type="text"/> Mini Flyweigh (-48kg)

**CONFIRM** **CANCEL**

Successfully

Your data inserted successfully

**OK**

You Successfully Registered in Muay Thai

\*\*\*\*\* Receipt \*\*\*\*\*  
\*\*\*\*\* Sports Martial Arts : Muay Thai \*\*\*\*\*  
  
NAME : NUR HIDAYAH  
AGE : 17  
GENDER : FEMALE  
NRIC : 060309070104  
EMAIL : dayah@gmail.com  
PHONE NUMBER : 018-94 9876  
WEIGHT : 48.98 kg  
CATEGORY : Mini Flyweigh (-48kg)  
\*\*\*\*\*

Your Registration Receipt

\*\*\* Receipt \*\*\*  
\*\*\* Sports Martial Arts : Muay Thai \*\*\*

Your Information :

NAME	:	NUR HIDAYAH
AGE	:	17
GENDER	:	FEMALE
NRIC	:	060309070104
E-MAIL	:	dayah@gmail.com
PHONE NUMBER	:	018-94 9876
WEIGHT	:	48.98 kg
CATEGORY	:	Mini Flyweigh (-48kg)

nama	age	nric	gender	email	phonenum	weight	category
NUR HIDAYAH	17	060309070104	FEMALE	dayah@gmail.com	018-94 9876	48.98 kg	Mini Flyweigh (-48kg)

SMARS/ SILAT.py:

SILAT clicked!

Silat Registration System

\*\*\*Welcome To Silat Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/>
NRIC	:	<input type="text"/> Example : 010203040506
E-MAIL	:	<input type="text"/>
PHONE NUMBER	:	<input type="text"/>
WEIGHT	:	<input type="text"/>

**SUBMIT** **RESET**

NAME : NAQIB KHUSAIRY  
AGE : 20  
GENDER : MALE  
NRIC : 030617104289  
EMAIL : Naqib@gmail.com  
PHONE NUMBER : 012-345 6789  
WEIGHT : 57.23 kg  
CATEGORY : Class C (56kg - 60kg)

Silat Registration System

\*\*\*Welcome To Silat Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> Naqib Khusairy
NRIC	:	<input type="text"/> 030617104289 Example : 010203040506
E-MAIL	:	<input type="text"/> Naqib@gmail.com
PHONE NUMBER	:	<input type="text"/> 012-345 6789
WEIGHT	:	<input type="text"/> 57229

**SUBMIT** **RESET**

Your Category :

Category	:	Class C (56kg - 60kg)
----------	---	-----------------------

Silat Registration System

\*\*\*Welcome To Silat Registration System\*\*\*

Fill Your Information Here :

NAME	:	<input type="text"/> Naqib Khusairy
NRIC	:	<input type="text"/> 030617-10-4289 Example : 010203040506
E-MAIL	:	<input type="text"/> Naqib@gmail.com
PHONE NUMBER	:	<input type="text"/> 012-345 6789
WEIGHT	:	<input type="text"/> 57229

**SUBMIT** **RESET**

Please Confirm Your Data

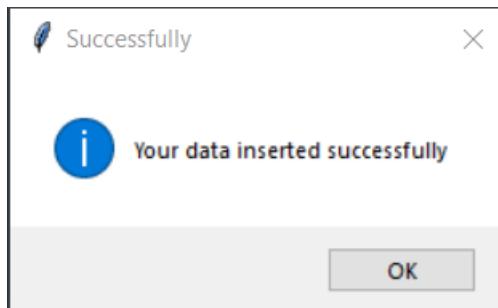
Confirm Your Data

\*\*\* Please Confirm Your Data \*\*\*  
Sports Martial Arts : Silat

Your Information :

NAME	:	<input type="text"/> NAQIB KHUSAIRY
AGE	:	<input type="text"/> 20
GENDER	:	<input type="text"/> MALE
NRIC	:	<input type="text"/> 030617104289
E-MAIL	:	<input type="text"/> Naqib@gmail.com
PHONE NUMBER	:	<input type="text"/> 012-345 6789
WEIGHT	:	<input type="text"/> 57.23 kg
CATEGORY	:	<input type="text"/> Class C (56kg - 60kg)

**CONFIRM** **CANCEL**



You Successfully Registered in Silat

\*\*\*\*\* Receipt \*\*\*\*\*  
\*\*\*\*\* Sports Martial Arts : Silat \*\*\*\*\*

NAME : NAQIB KHUSAIRY  
AGE : 20  
GENDER : MALE  
NRIC : 030617104289  
EMAIL : Naqib@gmail.com  
PHONE NUMBER : 012-345 6789  
WEIGHT : 57.23 kg  
CATEGORY : Class C (56kg - 60kg)

\*\*\*\*\*

Your Registration Receipt

\*\*\* Receipt \*\*\*  
\*\*\* Sports Martial Arts : Silat \*\*\*

Your Information :

NAME	:	NAQIB KHUSAIRY
AGE	:	20
GENDER	:	MALE
NRIC	:	030617104289
E-MAIL	:	Naqib@gmail.com
PHONE NUMBER	:	012-345 6789
WEIGHT	:	57.23 kg
CATEGORY	:	Class C (56kg - 60kg)

nama	age	nric	gender	email	phonenum	weight	category
NAQIB KHUSAIRY	20	030617104289	MALE	Naqib@gmail.com	012-345 6789	57.23 kg	Class C (56kg - 60kg)

SMARS/ TAEKWONDO.py:

TAEKWONDO clicked!

Taekwondo Registration System

\*\*\*Welcome To Taekwondo Registration System\*\*\*

Fill Your Information Here :—

NAME	:	<input type="text"/>
NRIC	:	<input type="text"/> Example : 010203040506
E-MAIL	:	<input type="text"/>
PHONE NUMBER	:	<input type="text"/>
WEIGHT	:	<input type="text"/>

SUBMIT    RESET

NAME : SITI HAWA  
AGE : 21  
GENDER : FEMALE  
NRIC : 021020051002  
EMAIL : hawa@gmail.com  
PHONE NUMBER : 011 5557 1002  
WEIGHT : 53.79 kg  
CATEGORY : Fin (-54kg)

Taekwondo Registration System

\*\*\*Welcome To Taekwondo Registration System\*\*\*

Fill Your Information Here :—

NAME	:	<input type="text"/> Siti Hawa
NRIC	:	<input type="text"/> 021020051002 Example : 010203040506
E-MAIL	:	<input type="text"/> hawa@gmail.com
PHONE NUMBER	:	<input type="text"/> 011 5557 1002
WEIGHT	:	<input type="text"/> 53789

SUBMIT    RESET

Your Category :—

Category	:	Fin (-54kg)
----------	---	-------------

Taekwondo Registration System

\*\*\*Welcome To Taekwondo Registration System\*\*\*

Fill Your Information Here :—

NAME	:	<input type="text"/> Siti Hawa
NRIC	:	<input type="text"/> 021020051002 Example : 010203040506
E-MAIL	:	<input type="text"/> hawa@gmail.com
PHONE NUMBER	:	<input type="text"/> 011 5557 1002
WEIGHT	:	<input type="text"/> 53789

SUBMIT    RESET

Please Confirm Your Data

Confirm Your Data

\*\*\* Please Confirm Your Data \*\*\*  
Sports Martial Arts : Taekwondo

Your Information :—

NAME	:	<input type="text"/> SITI HAWA
AGE	:	<input type="text"/> 21
GENDER	:	<input type="text"/> FEMALE
NRIC	:	<input type="text"/> 021020051002
E-MAIL	:	<input type="text"/> hawa@gmail.com
PHONE NUMBER	:	<input type="text"/> 011 5557 1002
WEIGHT	:	<input type="text"/> 53.79 kg
CATEGORY	:	<input type="text"/> Fin (-54kg)

CONFIRM    CANCEL

Successfully



Your data inserted successfully

OK

You Successfully Registered in Taekwondo

\*\*\*\*\* Receipt \*\*\*\*\*

\*\*\*\*\* Sports Martial Arts : Taekwondo \*\*\*\*\*

NAME : SITI HAWA  
AGE : 21  
GENDER : FEMALE  
NRIC : 021020051002  
EMAIL : hawa@gmail.com  
PHONE NUMBER : 011 5557 1002  
WEIGHT : 53.79 kg  
CATEGORY : Fin (-54kg)

\*\*\*\*\*

Your Registration Receipt

\*\*\* Receipt \*\*\*  
\*\*\* Sports Martial Arts : Taekwondo \*\*\*

Your Information :

NAME	:	SITI HAWA
AGE	:	21
GENDER	:	FEMALE
NRIC	:	021020051002
E-MAIL	:	hawa@gmail.com
PHONE NUMBER	:	011 5557 1002
WEIGHT	:	53.79 kg
CATEGORY	:	Fin (-54kg)

nama	age	nric	gender	email	phonenum	weight	category
SITI HAWA	21	021020051002	FEMALE	hawa@gmail.com	011 5557 1002	53.79 kg	Fin (-54kg)