



المدرسة العليا
للتكنولوجيا - الصويرة
L'ÉCOLE SUPÉRIEURE DE
TECHNOLOGIE – ESSAOUIRA

Ecole Supérieure de Technologie
Essaouira

Compte Rendu Mini Projet d'Oracle
Bases de Données Avancées
Informatique Décisionnelle et Science de Données

Réalisé par :

Salah Eddine ZKARA.

Yassine HANNAFI.

Encadrant :

Pr. Lamia

ZIAD

Année universitaire :

2020/2021

Sommaire

INTRODUCTION	3
I- CONNEXION AU BASE DE DONNES	4
II- CRÉATION DES TABLES	5
III- MANIPULATION DES DONNES	6
1-ADMIN DASHBOARD.	6
A-AJOUTER.	7
B-SUPPRIMER.	7
2-ELECTION DASHBOARD.	8
A-CRÉATION DU COMPTE.	8
B-AUTHENTIFICATION/VOTE.	9
C-STATISTIQUES.	11
CONCLUSION	12

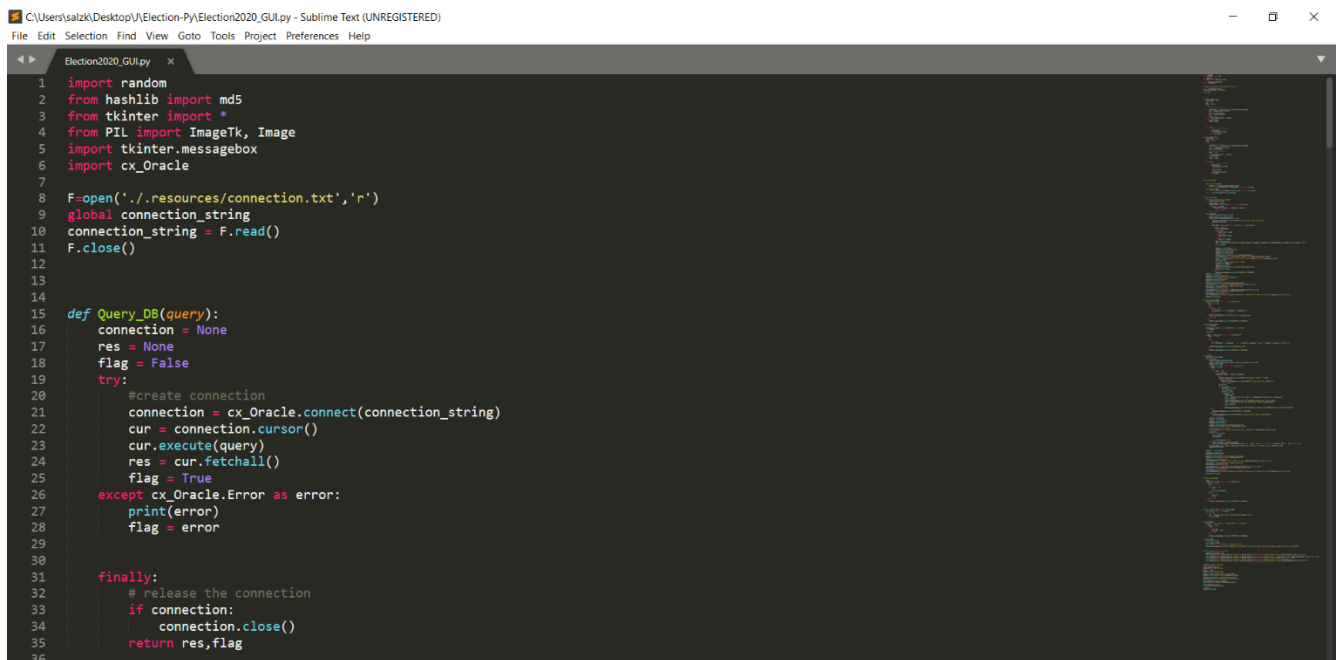
Introduction

Le but de ce Mini Projet du module de base de données avancées est la mise en place d'une petite application d'élection en utilisant Oracle (SQL) pour la gestion de la base de données et le langage python pour interagir avec l'utilisateur.

I- Connexion au Base de donnes

Pour se connecter à la base de donnes on va utiliser "cx_Oracle" un module Python qui permet de se connecter à une base de données Oracle à partir de Python.

- Tout d'abord il faut avoir oracle installe sur notre système.
- Il faut entrer les informations de connexion dans le fichier "./.resources/connection.txt" comme suite : 'username/password@localhost/YourTns'
- On crée un Objet de connexion avec les informations de connexion.
- On crée un Objet de curseur à l'aide d'objet de connexion précédent.
- Maintenant on peut exécuter les requêtes de SQL a l'aide de notre curseur.
- On n'oublie pas de fermer la connexion après avoir terminé l'exécution des requetés.

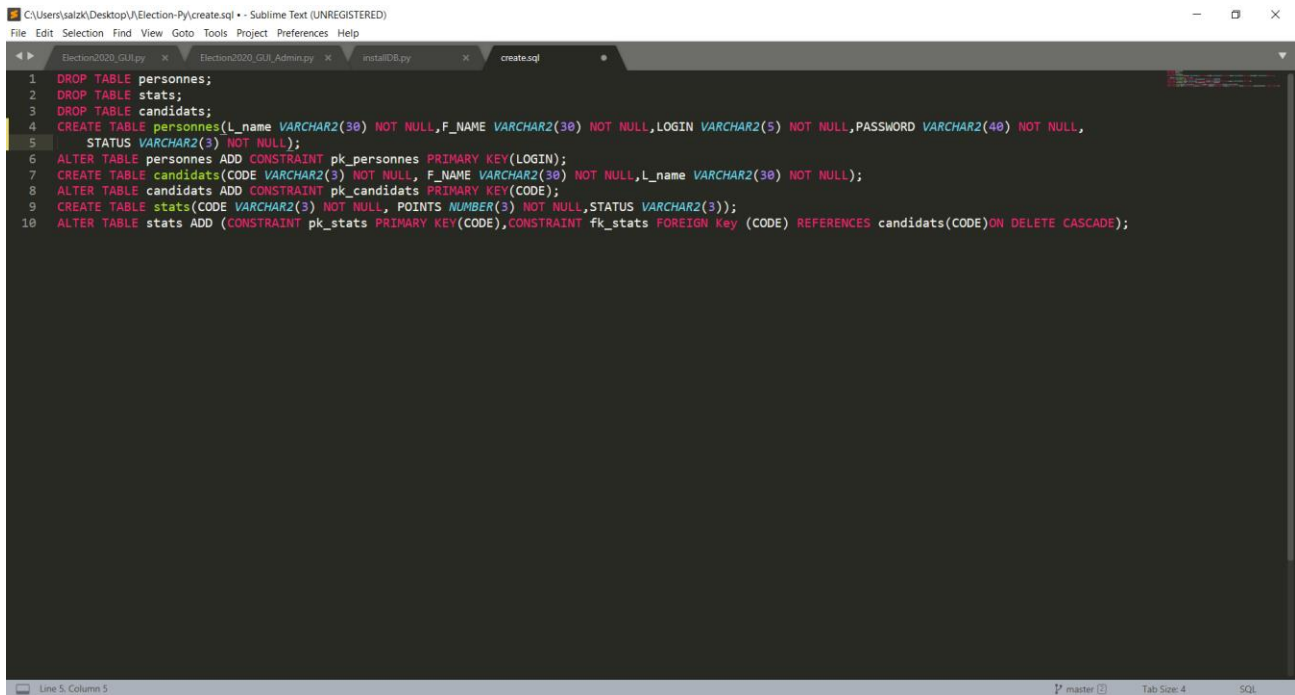


```
C:\Users\sakzi\Desktop\Election-Py\Election2020_GUI.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Election2020_GUI.py x
1 import random
2 from hashlib import md5
3 from tkinter import *
4 from PIL import ImageTk, Image
5 import tkinter.messagebox
6 import cx_Oracle
7
8 f=open('./.resources/connection.txt','r')
9 global connection_string
10 connection_string = f.read()
11 f.close()
12
13
14
15 def Query_DB(query):
16     connection = None
17     res = None
18     flag = False
19     try:
20         #create connection
21         connection = cx_Oracle.connect(connection_string)
22         cur = connection.cursor()
23         cur.execute(query)
24         res = cur.fetchall()
25         flag = True
26     except cx_Oracle.Error as error:
27         print(error)
28         flag = error
29
30
31 finally:
32     # release the connection
33     if connection:
34         connection.close()
35     return res,flag
36
```

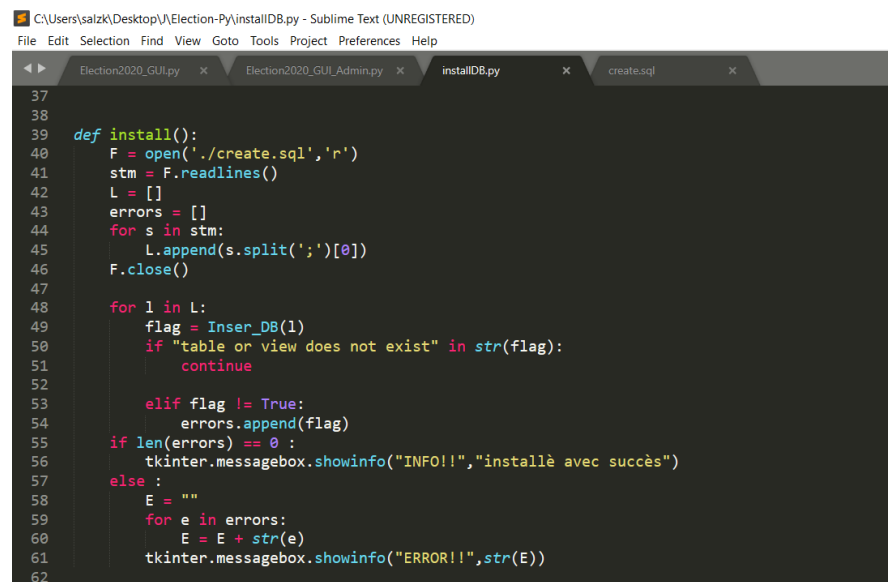
II- Création des tables

Pour créer les tables nécessaires il faut exécuter ces commandes dans le SQL développer ou SQL plus.

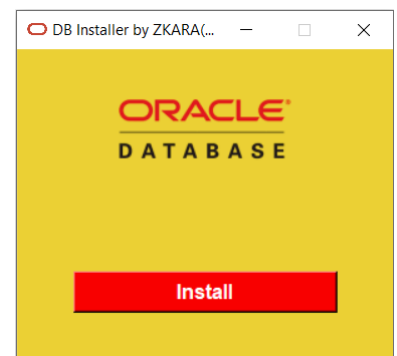


```
1 DROP TABLE personnes;
2 DROP TABLE stats;
3 DROP TABLE candidats;
4 CREATE TABLE personnes(L_name VARCHAR2(30) NOT NULL,F_NAME VARCHAR2(30) NOT NULL,LOGIN VARCHAR2(5) NOT NULL,PASSWORD VARCHAR2(40) NOT NULL,
5 STATUS VARCHAR2(3) NOT NULL);
6 ALTER TABLE personnes ADD CONSTRAINT pk_personnes PRIMARY KEY(LOGIN);
7 CREATE TABLE candidats(CODE VARCHAR2(3) NOT NULL, F_NAME VARCHAR2(30) NOT NULL,L_name VARCHAR2(30) NOT NULL);
8 ALTER TABLE candidats ADD CONSTRAINT pk_candidats PRIMARY KEY(CODE);
9 CREATE TABLE stats(CODE VARCHAR2(3) NOT NULL, POINTS NUMBER(3) NOT NULL,STATUS VARCHAR2(3));
10 ALTER TABLE stats ADD (CONSTRAINT pk_stats PRIMARY KEY(CODE),CONSTRAINT fk_stats FOREIGN Key (CODE) REFERENCES candidats(CODE)ON DELETE CASCADE);
```

Mais pour améliorer cela pour un utilisateur normal on va créer une petite interface avec un bouton "Install" pour exécuter les commandes ligne par lignes et les tables seront créer.



```
37
38
39 def install():
40     F = open('./create.sql','r')
41     stm = F.readlines()
42     L = []
43     errors = []
44     for s in stm:
45         L.append(s.split(';')[0])
46     F.close()
47
48     for l in L:
49         flag = Insert_DB(l)
50         if "table or view does not exist" in str(flag):
51             continue
52
53         elif flag != True:
54             errors.append(flag)
55     if len(errors) == 0 :
56         tkinter.messagebox.showinfo("INFO!!","installé avec succès")
57     else :
58         E = ""
59         for e in errors:
60             E = E + str(e)
61         tkinter.messagebox.showinfo("ERROR!!",str(E))
62
```

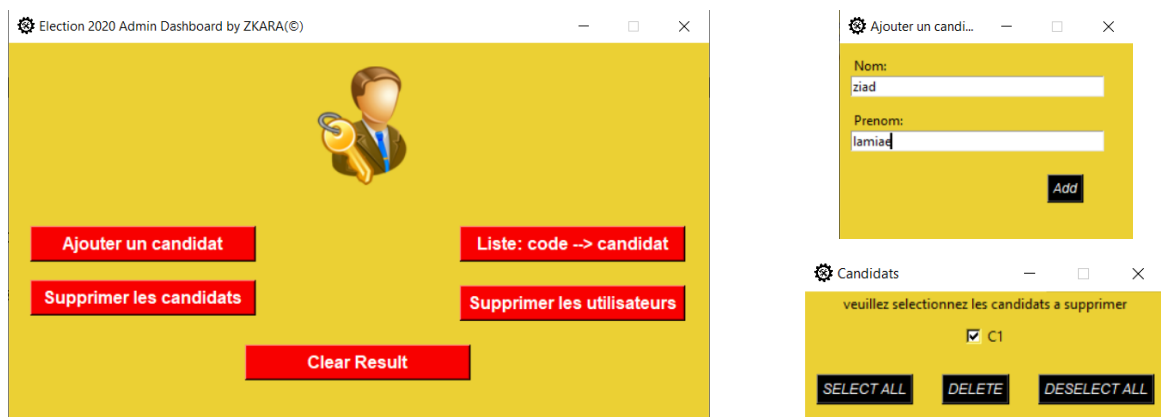


Après qu'on click sur le bouton "Install" les requêtes de création vont être exécuter l'une après l'autre et au cas où les tables ont déjà été créer, ils vont être supprimer et recréer à zéro.

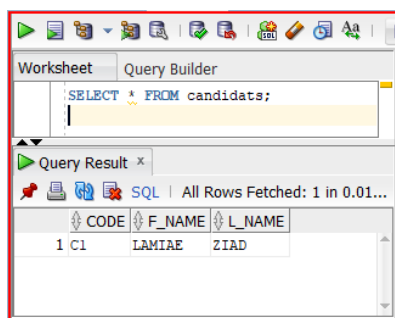
III- Manipulation des donnees

1-Admin Dashboard.

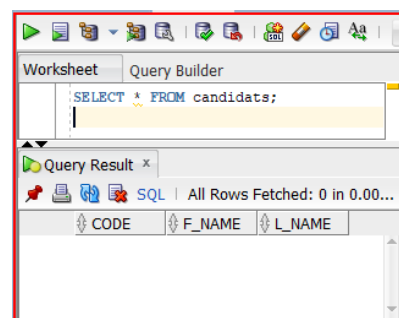
Cette Interface est réservée pour l'administrateur qui pourra ajouter ou supprimer les candidats ou les utilisateurs.



Voilà le candidat a été ajoute dans notre table.



Voilà le candidat a été supprimer de la table.



a-Ajouter.

Pour ajouter un candidat. (Le code source complet est disponible sur mon compte [GitHub](#))

```
C:\Users\sakzi\Desktop\Election-Py\Election2020_GUI\Admin.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Election2020_GUI.py x Election2020_GUI_Admin.py x installDB.py x create.sql x

70 def click():
71     lname=textentry1.get().upper()
72     fname=textentry2.get().upper()
73     if(Inscription(Lname,Fname)==False):
74         tkinter.messagebox.showinfo("ALERT!!","Ce candidat est déjà enregistré!")
75         window2.destroy()
76     else:
77         num,flag = Query_DB("SELECT COUNT(*) FROM candidats")
78         if flag == True:
79             i = int(num[0][0])+1
80             stm = f"INSERT INTO candidats VALUES('C{str(i)}','{Fname}','{Lname}')"
81             flag = Inser_DB(stm)
82             if flag == True:
83                 tkinter.messagebox.showinfo("INFO!!","ajouté avec succès")
84                 clear_()
85             else:
86                 tkinter.messagebox.showinfo("ERROR!!",str(flag))
87         else:
88             tkinter.messagebox.showinfo("ERROR!!",str(flag))
89         window2.destroy()
90
91 window2 =Toplevel()
92 window2.title("Ajouter un candidat")
93 window2.configure(background="#ebd834")
94 window2.geometry("270x180")
95 window2.resizable(0, 0)
96 window2.iconbitmap('./resources/gear.ico')
97 Label(window2,text="Nom:",bg=background_color).place(x=10,y=10)
98 textentry1= Entry(window2,bg="white",width=38)
99 textentry1.place(x=10,y=30)
100 Label(window2,text="Prenom:",bg=background_color).place(x=10,y=60)
101 textentry2= Entry(window2,bg="white",width=38)
102 textentry2.place(x=10,y=80)
103 Button(window2,font="none 9 italic",fg="white",bg="black",text="Add",command=click).place(x=190,y=120)
104 window2.mainloop()
105
106
107 def Liste_Candidat():
```

b-Supprimer.

Pour supprimer un candidat

```
C:\Users\sakzi\Desktop\Election-Py\Election2020_GUI\Admin.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

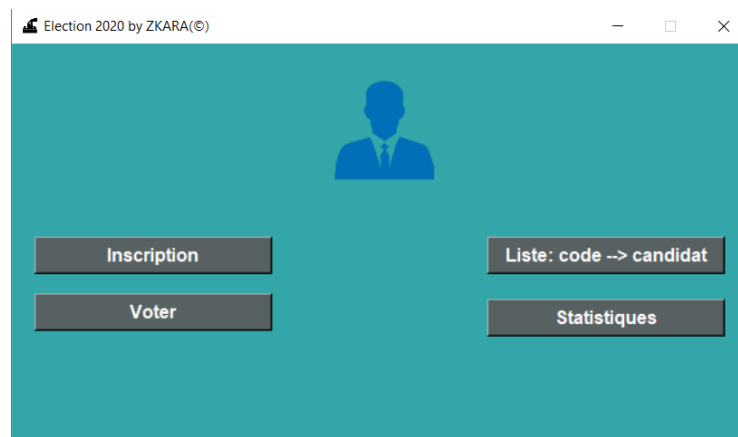
Election2020_GUI.py x Election2020_GUI_Admin.py x installDB.py x create.sql x

121 def Delete_cand():
122     def click():
123         i = 0
124         j = 0
125         for e in D:
126             if D[e].get() == 1:
127                 j += 1
128                 stm = f"DELETE FROM candidats WHERE code = '{e}' "
129                 flag = Inser_DB(stm)
130                 if flag != True:
131                     i += 1
132                     tkinter.messagebox.showinfo("ERROR!!",str(flag))
133             j = j - i
134             tkinter.messagebox.showinfo("ALERT!!",str(j)+" candidats supprimer avec succès")
135             clear_()
136             window5.destroy()
137
138 def sel_all():
139     for a in A:
140         a.select()
141 def dsel_all():
142     for a in A:
143         a.deselect()
144
145 n=Liste_Candidat()
146 window5 = Toplevel()
147 window5.resizable(0, 0)
148 window5.title("Candidats")
149 window5.iconbitmap('./resources/gear.ico')
150 window5.configure(background=background_color)
151 Label(window5,text="veuillez selectionnez les candidats a supprimer",bg=background_color).pack()
152 D = {}
153 A = []
154
155 text,flag = Query_DB("SELECT code FROM candidats")
156 if flag == True:
157     i = 0
158     for a in text:
```

(Même chose pour la gestion des utilisateurs).

2-Election Dashboard.

Cette Interface est réservée pour l'utilisateur qui peut voter sur un seul candidat une seule fois.

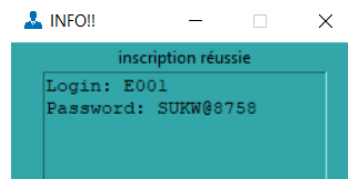


a-Création du compte.

Nom:
zkara

Prenom:
salah eddine

Sign Up



Un code et un mot de passe sont attribué aléatoirement à l'utilisateur.

Voilà l'utilisateur a été ajouté dans la table « personne ».

Worksheet Query Builder

SELECT * FROM personnes;

Query Result x

SQL All Rows Fetched: 1 in 0.003 seconds

	L_NAME	F_NAME	LOGIN	PASSWORD	STATUS
1	ZKARA	SALAH EDDINE	E001	b01bf8dcb724a98d59c295abdc72a0bf	OK

Le mot de passe est sécurisé à l'aide du cryptage « md5 »


```

C:\Users\sakzi\Desktop\J\Election-Py\Election2020_GUI.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

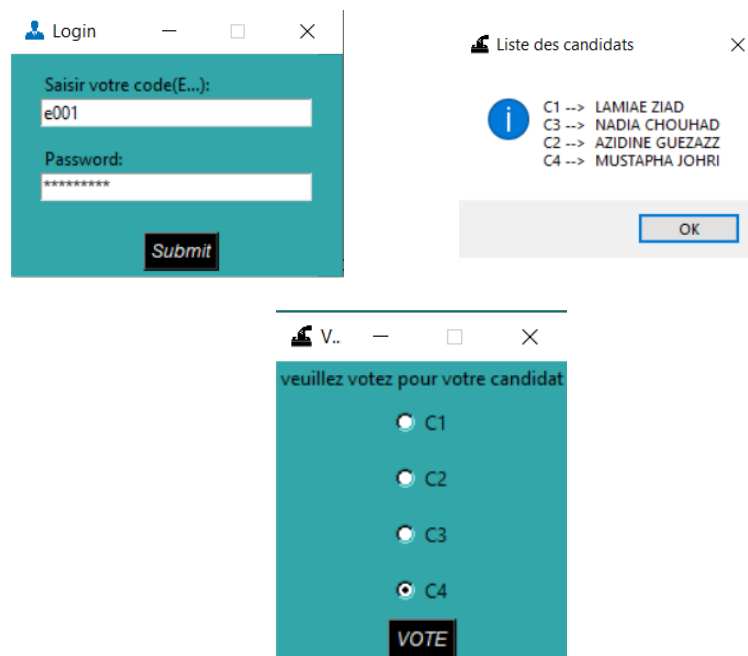
Election2020_GUI.py Election2020_GUI_Admin.py instalDB.py createad

72 def Inscription():
73     def Inscription_(Lname,Fname):
74         Lname=Lname.upper()
75         Fname=Fname.upper()
76         result,flag = Query_DB("SELECT * FROM personnes")
77         for line in result:
78             if (line[0]==Lname and (line[1]==Fname):
79                 return False
80             return True
81     def click():
82         Lname=textentry1.get().upper()
83         Fname=textentry2.get().upper()
84         if(Inscription_(Lname,Fname)==False) :
85             tkinter.messagebox.showinfo("ALERTE!!","Vous etes deja inscrit!")
86             window2.destroy()
87         else:
88             num,flag = Query_DB("SELECT COUNT(*) FROM personnes")
89             if flag == True:
90                 i=int(num[0][0])+1
91                 if(i<10) :
92                     code='E00'+str(i)
93                 elif(i<100):
94                     code='E0'+str(i)
95                 else:
96                     code='E'+str(i)
97                 paswd=randomPass()
98                 stm = f"INSERT INTO personnes VALUES('{Lname}','{Fname}','{code}','{str(md5(paswd.encode()).hexdigest())}','OK')"
99                 Inser_DB(stm)
100
101             window4= Toplevel()
102             window4.geometry("250x100")
103             window4.title("INFO!!")
104             window4.resizable(0,0)
105             window4.configure(background=background_color)
106             Label(window4,text="inscription réussie",bg=background_color).pack()
107             output= Text(window4,width=25,height=10,wrap=WORD,background=background_color)
108             output.pack()
109             m="Login: "+code+"\nPassword: "+paswd

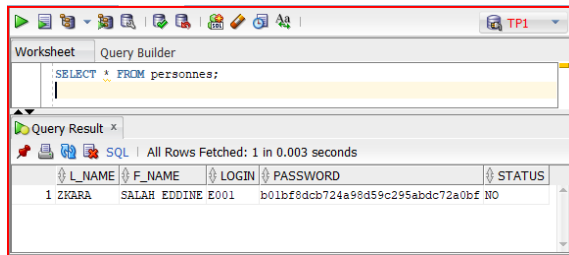
```

b-Authentication/Vote.

On va utiliser l'utilisateur qu'on a créé dans l'étape précédente.



Voilà l'utilisateur qu'on a utilisé n'a plus le droit de voter, jusqu'à ce que l'administrateur mis à zéro les résultats.



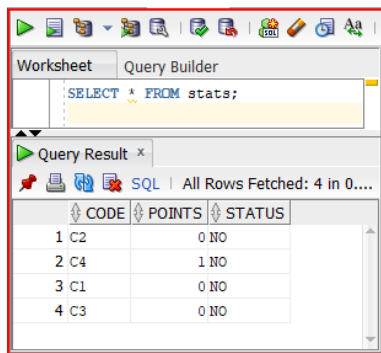
Query Builder: `SELECT * FROM personnes;`

Query Result: All Rows Fetched: 1 in 0.003 seconds

	L_NAME	F_NAME	LOGIN	PASSWORD	STATUS
1	ZKARA	SALAH EDDINE	E001	b01bf8dcb724a98d59c295abdc72a0bf	NO

STATUS : NO

Et voilà les points du candidat qu'on a voté ont été incrémenter par 1.

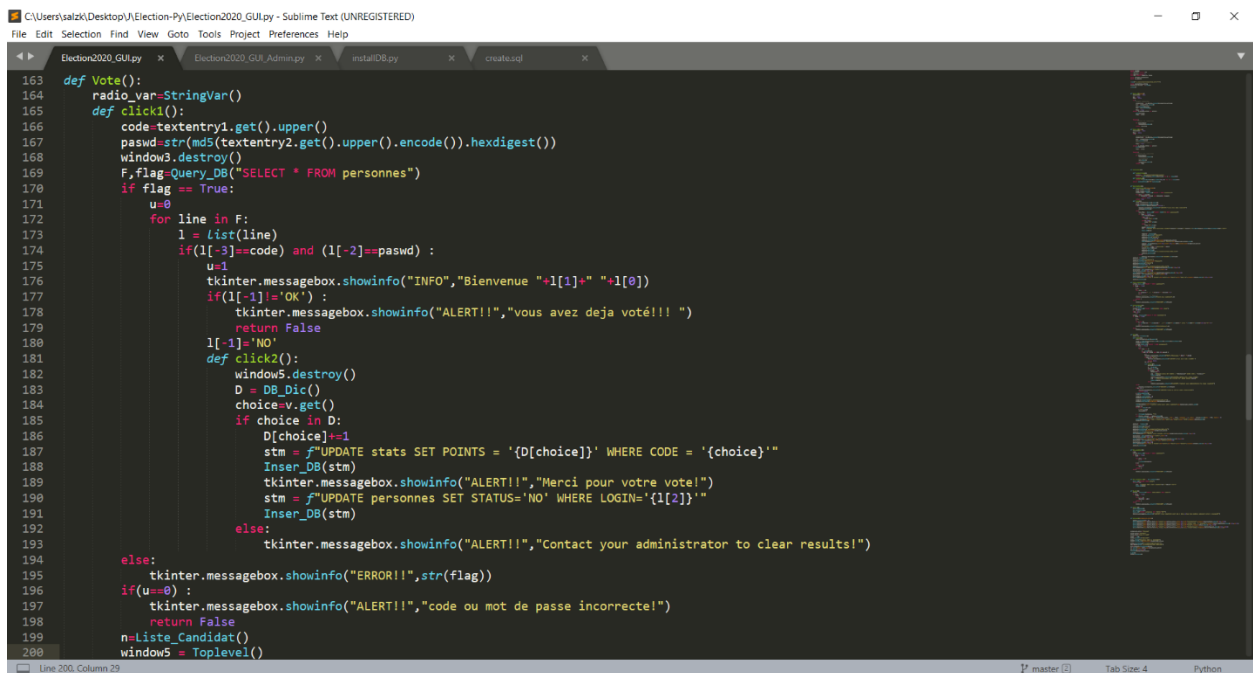


Query Builder: `SELECT * FROM stats;`

Query Result: All Rows Fetched: 4 in 0....

	CODE	POINTS	STATUS
1	C2	0	NO
2	C4	1	NO
3	C1	0	NO
4	C3	0	NO

(Je rappel que le code source complet est disponible sur mon compte [GitHub](#)).



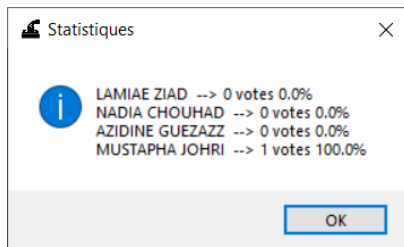
```

163 def Vote():
164     radio_var=StringVar()
165     def click1():
166         code=textentry1.get().upper()
167         paswd=str(md5(textentry2.get().upper().encode()).hexdigest())
168         window3.destroy()
169         F_flag_Query_DB("SELECT * FROM personnes")
170         if flag == True:
171             u=0
172             for line in F:
173                 l = List(line)
174                 if (l[-3]==code) and (l[-2]==paswd) :
175                     u=1
176                     tkinter.messagebox.showinfo("INFO","Bienvenue "+l[1]+" "+l[0])
177                     if (l[-1]!='OK') :
178                         tkinter.messagebox.showinfo("ALERT!!","vous avez deja voté!!! ")
179                     return False
180                 l[-1]='NO'
181             def click2():
182                 window5.destroy()
183                 D = DB_Dic()
184                 choice=v.get()
185                 if choice in D:
186                     D[choice]+=1
187                     stm = f"UPDATE stats SET POINTS = '{D[choice]}' WHERE CODE = '{choice}'"
188                     Inser_DB(stm)
189                     tkinter.messagebox.showinfo("ALERT!!","Merci pour votre vote!")
190                     stm = f"UPDATE personnes SET STATUS='NO' WHERE LOGIN='{l[2]}'"
191                     Inser_DB(stm)
192                 else:
193                     tkinter.messagebox.showinfo("ALERT!!","Contact your administrator to clear results!")
194             else:
195                 tkinter.messagebox.showinfo("ERROR!!",str(flag))
196             if (u==0) :
197                 tkinter.messagebox.showinfo("ALERT!!","code ou mot de passe incorrecte!")
198             return False
199         n=Liste_Candidat()
200         window5 = Toplevel()

```

c-Statistiques.

Voilà les statistiques ont été affiche.



Le code source.

```
C:\Users\sakzi\Desktop\Election-Py\Election2020_GUI.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Election2020_GUI.py Election2020_GUI_Admin.py installDB.py create.sql

145 def Statistiques():
146     D = DB_Dic()
147     n, flag = Query_DB("SELECT SUM(POINTS) FROM stats")
148     n = n[0][0]
149     if (n==0) :
150         n=1
151     A, flag = Query_DB("SELECT * FROM candidats")
152     if flag == True:
153         i=1
154         R=""
155         for a in A:
156             R+=str(a[1])+" "+str(a[2])+" --> "+str(D['C'+str(i)])+" votes "+str(D['C'+str(i)]*100/n)+"%"+'\n'
157             i+=1
158         tkinter.messagebox.showinfo("Statistiques",R)
159     else:
160         tkinter.messagebox.showinfo("ERROR!!",str(flag))
161
162
163 def Vote():
164     radio_var=StringVar()
165     def click1():
166         code=textentry1.get().upper()
167         paswd=str(md5(textentry2.get().upper().encode()).hexdigest())
168         window3.destroy()
169         F, flag = Query_DB("SELECT * FROM personnes")
170         if flag == True:
171             u=0
172             for line in F:
173                 l = list(line)
174                 if (l[-3]==code and (l[-2]==paswd) :
175                     u=1
176                     tkinter.messagebox.showinfo("INFO", "Bienvenue "+l[1]+" "+l[0])
177                     if (l[-1]!='OK') :
178                         tkinter.messagebox.showinfo("ALERT!!", "vous avez deja voté!!! ")
179                     return False
180                 l[-1]='NO'
181             def click2():
182                 window5.destroy()
```

[Code Source Complet.](#)

Conclusion

Dans ce Mini Projet on a pratiqué des notions du LDD et LMD du langage SQL acquises durant la séance du cours et TP.