**RAPPORT ECRIT**

1. **Membres du groupe**

HOUESSOU Crépin

GANDONOU Shalom

AZA-GNANDJI Juste

SONON Christyl

**SOMMAIRE**

Modélisation de la base de données (SLR)…………………**3**

Création de la base de données avec requête……………**3**

Indications d’utilisation………………………………………………**3**

Création interface avec la bibliothèque **tkinter…………..3**

Page principale(main.py) ………………………………………..…..**4**

Page de traitement des étudiants (Page1.py) ………………**7**

Page de traitement des matières (Page2.py)………………..**11**

Page de traitements des évaluations (Page3.py)…………**15**

Page de traitements des notes (Page4.py)……………………**19**

Page de traitements des bulletins (Page5.py)…………………**27**

**Démarche**

-Modélisation de la base de données (SLR)

Etudiants (idetudiants,nom)

Matieres (idmatieres, NomMat, CoefMat)

Evaluations (idevaluation, Nomevaluation, Coefficient)

Composer (idetudiants, idmatieres, idevaluation, note)

-Création de la base de données avec requête.

CREATE DATABASE ecole

CREATE TABLE etudiants (idetudiants INT PRIMARY KEY AUTO\_INCREMENT, nom VARCHAR (255))

CREATE TABLE matieres (idmatieres INT PRIMARY KEY AUTO\_INCREMENT,

NomMat VARCHAR (255), CoefMat INT)

CREATE TABLE evaluations (idevaluation INT PRIMARY KEY AUTO\_INCREMENT, Nomevaluation VARCHAR (255), Coefficient INT)

CREATE TABLE composer (idetudiants INT, idmatieres INT NOT NULL, idevaluation INT NOT NULL, note float, CONSTRAINT fk\_etudiants FOREIGN KEY (idetudiants) REFERENCES etudiants(idetudiants), CONSTRAINT fk\_matieres FOREIGN KEY (idmatieres) REFERENCES matieres(idmatieres), CONSTRAINT fk\_evaluations FOREIGN KEY (idevaluation) REFERENCES evaluations(idevaluation), CONSTRAINT pk\_cle PRIMARY KEY (idetudiants, idmatieres,idevaluation))

-Indications d’utilisation

Veuillez copier et exécuter la requête de création de la base de données (CREATE DATABASE ecole).

Veuillez importer le fichier (ecole.sql) dans la base de données précédemment

créée.

Dans le cas ou le fichier(pythonprojet>output>main>main.exe) ne s’exécuterait pas correctement, veuillez exécuter le fichier (pythonprojet>main.py)

-Création interface avec la bibliothèque **tkinter.**

* Page principale(main.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#Cette page permet de naviguer entre les différentes pages de l'application

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("1366x768+0+0")

root.resizable(False, False)

root.configure(background="#CDC8B1")

def add\_etudiant():

    try:

        call(["python", "page1.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def add\_matieres():

    try:

        call(["python", "page2.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def add\_evaluations():

    try:

        call(["python", "page3.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def add\_notes():

    try:

        call(["python", "page4.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def add\_bul():

    try:

        call(["python", "page5.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

#creation d'une barre de menu

menu\_bar = Menu(root)

#creer un premier menu

file\_menu  = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Ajout des etudiants", menu=file\_menu)

file\_menu.add\_command(label="ajouter",command=add\_etudiant)

#creer un deuxieme menu

file\_menu  = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Ajout des matieres", menu=file\_menu)

file\_menu.add\_command(label="ajouter",command=add\_matieres)

#creer un troisieme menu

file\_menu  = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Ajout des evaluations", menu=file\_menu)

file\_menu.add\_command(label="ajouter",command=add\_evaluations)

#creer un quatrieme menu

file\_menu  = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Ajout des notes", menu=file\_menu)

file\_menu.add\_command(label="ajouter",command=add\_notes)

#creer un cinquieme menu

file\_menu  = Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Bulletin", menu=file\_menu)

file\_menu.add\_command(label="ajouter",command=add\_bul)

#configurer notre fenetre pour ajouter cette menu bar

root.config(menu=menu\_bar)

#execution

root.mainloop()

* Page de traitement des étudiants (Page1.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#ETUDIANT\_AJOUT\_MODIFIER\_SUPPRIMER

def ajouter():

    nom = txtnom.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="INSERT INTO etudiants (Nom) VALUES(%s)"

        val =(nom,)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("information", "bien ajouter")

        root.destroy()

        call(["python", "page1.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def modifier():

    nom = txtnom.get()

    numero = txtidetud.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="update etudiants set Nom=%s where idetudiants= %s"

        val =(nom,numero)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "modifiée")

        root.destroy()

        call(["python", "page1.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def supprimer():

    numero = txtidetud.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="delete from etudiants where idetudiants=%s"

        val =(numero,)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "supprimée")

        root.destroy()

        call(["python", "page1.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("1120x400+0+0")

root.resizable(False, False)

root.configure(background="silver")

#Ajouter le titre

lbltitre = Label(root,borderwidth = 3,relief = GROOVE,text = "ENREGISTREMENT DES ETUDIANTS",

font = ("sans serif", 25), background = "silver",fg="#fffafa")

lbltitre.place(x=0,y=0,width= 1350, height=100)

#details des inscriptions

#idetudiants

lblidetud = Label (root, text="IDETUDIANTS", font=("Arial", 10),bg="white", fg="#000")

lblidetud.place(x=70, y=110,width=150)

txtidetud = Entry(root, bd=4, font=("Arial", 10))

txtidetud.place(x=250,y=110,width=300)

#nom

lblnom = Label (root, text="NOM", font=("Arial", 10),bg="white", fg="#000")

lblnom.place(x=70, y=140,width=150)

txtnom = Entry(root, bd=4, font=("Arial", 10))

txtnom.place(x=250,y=140,width=300)

#AJOUTER

btnajouter = Button(root, text="AJOUTER", font=("Arial", 10),bg="#CDB79E",fg="white", command=ajouter)

btnajouter.place(x=250,y=230,width=85)

#modifier

btnmodifier = Button(root, text="MODIFIER", font=("Arial", 10), bg="#8A2BE2",fg="white", command=modifier)

btnmodifier.place(x=350, y=230, width=85)

#SUPPRIMER

btnsupprimer = Button(root, text="SUPPRIMER", font=("Arial", 10), bg="#DEB887",fg="white", command=supprimer)

btnsupprimer.place(x=450, y=230, width=85)

#TABLE\_ETUDIANTS

table = ttk.Treeview(root, columns = (1, 2, 3), height = 5, show = "headings")

table.place(x=560,y=110, height=260)

#Entete

table.heading(1, text = "NUM")

table.heading(2, text = "NOM")

table.heading(3, text = "PRENOM")

#dimentions pour colonnes

table.column(1,width = 50)

table.column(2,width = 200)

table.column(3,width = 300)

#Connexion\_etudiants

maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

meconnect = maBase.cursor()

meconnect.execute("select \* from etudiants")

for row in meconnect:

    table.insert('', END, value = row)

maBase.close()

#execution

root.mainloop()

* Page de traitement des matières (Page2.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#MATIERES\_AJOUT\_MODIFIER\_SUPPRIMER

def ajouter\_ma():

    nom = txtnomm.get()

    coefficient = combocoefmat.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="INSERT INTO matieres (NomMat,CoefMat) VALUES(%s,%s)"

        val =(nom,coefficient)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "bien ajoutée")

        root.destroy()

        call(["python", "page2.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def modifier\_ma():

    nom = txtnomm.get()

    numero = txtidmat.get()

    coefficient = combocoefmat.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="update matieres set NomMat=%s,CoefMat=%s where idmatieres= %s"

        val =(nom,coefficient,numero)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "modifiée")

        root.destroy()

        call(["python", "page2.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def supprimer\_ma():

    numero = txtidmat.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="delete from matieres where idmatieres=%s"

        val =(numero,)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "supprimée")

        root.destroy()

        call(["python", "page2.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("570x420+0+0")

root.resizable(False, False)

root.configure(background="#474747")

#Ajouter le titre

lbltitre = Label(root,borderwidth = 3,relief = GROOVE,text = "ENREGISTREMENT DES MATIERES",

font = ("sans serif", 18), background = "#8B7500",fg="#fffafa")

lbltitre.place(x=0,y=0,width= 600, height=60)

#details des inscriptions

#id\_matiere

lblidmat = Label (root, text="IDMATIERES", font=("Arial", 10),bg="#474747",fg="#fffafa")

lblidmat.place(x=70, y=100,width=150)

txtidmat = Entry(root, bd=4, font=("Arial", 10))

txtidmat.place(x=250,y=100,width=300)

#nom\_matiere

lblnomm = Label (root, text="NOM MATIERE", font=("Arial", 10),bg="#474747",fg="#fffafa")

lblnomm.place(x=70, y=130,width=150)

txtnomm = Entry(root, bd=4, font=("Arial", 10))

txtnomm.place(x=250,y=130,width=300)

#Coefficient matieres

lblcoefmat = Label(root,text="COEFFICIENT MATIERE", font=("Arial", 10),bg="#474747",fg="#fffafa")

lblcoefmat.place(x=70, y=160,width=150)

combocoefmat = ttk.Combobox(root,font=("Arial", 12))

combocoefmat['values'] = [1,2,3,4,5,6]

combocoefmat.place(x=250,y=160,width=300)

#AJOUTER\_matiere

btnajouter = Button(root, text="AJOUTER", font=("Arial", 10),bg="#CDB79E",fg="white",command=ajouter\_ma)

btnajouter.place(x=250,y=210,width=85)

#modifier\_matiere

btnmodifier = Button(root, text="MODIFIER", font=("Arial", 10), bg="#8A2BE2",fg="white",command=modifier\_ma)

btnmodifier.place(x=350, y=210, width=85)

#SUPPRIMER\_matiere

btnsupprimer = Button(root, text="SUPPRIMER", font=("Arial", 10), bg="#DEB887",fg="white",command=supprimer\_ma)

btnsupprimer.place(x=460, y=210, width=85)

#TABLE\_MATIERES

tablem = ttk.Treeview(root, columns = (1, 2), height = 2, show = "headings")

tablem.place(x=250,y=250, height=160)

#Entete

tablem.heading(1, text = "ID")

tablem.heading(2, text = "NOM")

#dimentions pour colonnes

tablem.column(1,width = 100)

tablem.column(2,width = 200)

#Connexion\_matieres

maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

meconnect = maBase.cursor()

meconnect.execute("select \* from matieres")

for row in meconnect:

    tablem.insert('', END, value = row)

maBase.close()

#execution

root.mainloop()

* Page de traitements des évaluations (Page 3.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#EVALUATIONS\_AJOUT\_MODIFIER\_SUPPRIMER

def ajouter\_eva():

    typ = txttype.get()

    coef = combocoef.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="INSERT INTO evaluations (Nomevaluation,Coefficient) VALUES(%s, %s)"

        val =(typ,coef)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "bien ajoutée")

        root.destroy()

        call(["python", "page3.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def modifier\_eva():

    typ = txttype.get()

    coef = combocoef.get()

    num = txtideval.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="update evaluations set Nomevaluation=%s, Coefficient=%s where idevaluation= %s"

        val =(typ,coef,num)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "modifiée")

        root.destroy()

        call(["python", "page3.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def supprimer\_eva():

    num = txtideval.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        sql ="delete from evaluations where idevaluation=%s"

        val =(num,)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "supprimée")

        root.destroy()

        call(["python", "page3.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

def nouveaute():

    try:

        call(["python", "page3.py"])

    except Exception as e:

        print(e)

        #retour

        maBase.rollback()

        maBase.close()

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("560x400+0+0")

root.resizable(False, False)

root.configure(background="#53868B")

#Ajouter le titre

lbltitre = Label(root,borderwidth = 3,relief = GROOVE,text = "ENREGISTREMENT DES EVALUATIONS",

font = ("sans serif", 15), background = "#53868B",fg="#FFF8DC")

lbltitre.place(x=0,y=0,width= 600, height=100)

#details des inscriptions

#AJOUT\_EVALUATIONS

#idevaluation

lblideval = Label (root, text="IDEVALUATION", font=("Arial", 10),bg="#53868B", fg="black")

lblideval.place(x=70, y=120,width=150)

txtideval = Entry(root, bd=4, font=("Arial", 10))

txtideval.place(x=250,y=120,width=300)

#typt\_evaluation

lbltype = Label (root, text="TYPE", font=("Arial", 10),bg="#53868B", fg="black")

lbltype.place(x=70, y=150,width=150)

txttype = Entry(root, bd=4, font=("Arial", 10))

txttype.place(x=250,y=150,width=300)

#COEFFICIENT

lblcoef = Label(root,text="COEFFICIENT", font=("Arial", 10),bg="#53868B",fg="black")

lblcoef.place(x=70, y=180,width=150)

combocoef = ttk.Combobox(root,font=("Arial", 12))

combocoef['values'] = [1,2,3,4,5,6]

combocoef.place(x=250,y=180,width=100)

#AJOUTER\_evaluations

btnajouter = Button(root, text="AJOUTER", font=("Arial", 10),bg="#53868B",fg="#FFF8DC",command=ajouter\_eva)

btnajouter.place(x=70,y=240,width=85)

#modifier\_matiere

btnmodifier = Button(root, text="MODIFIER", font=("Arial", 10), bg="#53868B",fg="#FFF8DC",command=modifier\_eva)

btnmodifier.place(x=160, y=240, width=85)

#SUPPRIMER\_matiere

btnsupprimer = Button(root, text="SUPPRIMER", font=("Arial", 10), bg="#53868B",fg="#8B3E2F",command=supprimer\_eva)

btnsupprimer.place(x=110, y=270, width=85)

#TABLE\_EVALUATIONS

tableev = ttk.Treeview(root, columns = (1, 2, 3), height = 5, show = "headings")

tableev.place(x=300,y=240, height=150)

#Entete

tableev.heading(1, text = "IDEVA")

tableev.heading(2, text = "TYPE")

tableev.heading(3, text = "COEF")

#dimentions pour colonnes

tableev.column(1,width = 50)

tableev.column(2,width = 150)

tableev.column(3,width = 50)

#Connexion\_evaluations

maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

meconnect = maBase.cursor()

meconnect.execute("select \* from evaluations")

for row in meconnect:

    tableev.insert('', END, value = row)

maBase.close()

#execution

root.mainloop()

* Page de traitements de notes (Page4.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#Fonction pour remplir les valeurs du combo box matieres

def cmb\_matiere():

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("select NomMat from matieres")

    data = []

    for row in meconnect.fetchall():

        data.append(row[0])

    return data

#Fonction pour remplir les valeurs du combo box composition

def cmb\_compo():

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("select Nomevaluation from evaluations")

    data = []

    for row in meconnect.fetchall():

        data.append(row[0])

    return data

#Fonction pour remplir les valeurs du combo box nom

def cmb\_nom():

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("select Nom from etudiants")

    data = []

    for row in meconnect.fetchall():

        data.append(row[0])

    return data

def etuajouter():

    nom = combonom.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor(buffered=True)

    try:

        pretusql = "SELECT idetudiants FROM etudiants WHERE Nom = %s"

        pretuval =(nom,)

        meconnect.execute(pretusql, pretuval)

        liste\_result = meconnect.fetchall()

        liste\_premier\_result = liste\_result[0]

        result = liste\_premier\_result[0]

        return result

        maBase.close()

    except Exception as e:

       print(e)

       #retour

       maBase.close()

def matajouter():

    mat = combomat.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor(buffered=True)

    try:

       prematsql = "SELECT idmatieres FROM matieres WHERE NomMat = %s"

       prematval = (mat,)

       meconnect.execute(prematsql, prematval)

       liste\_result = meconnect.fetchall()

       liste\_premier\_result = liste\_result[0]

       result = liste\_premier\_result[0]

       return result

       maBase.close()

    except Exception as e:

        print(e)

        maBase.close()

def compajouter():

    comp = combocomp.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor(buffered=True)

    try:

       preevalsql = "SELECT idevaluation FROM evaluations WHERE Nomevaluation = %s"

       preevalval = (comp,)

       meconnect.execute(preevalsql, preevalval)

       liste\_result = meconnect.fetchall()

       liste\_premier\_result = liste\_result[0]

       result = liste\_premier\_result[0]

       return result

       maBase.close()

    except Exception as e:

       print(e)

       #retour

       maBase.close()

def ajouter():

    notes = txtnotes.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor(buffered=True)

    try:

        resultprim = etuajouter()

        resultdeux = matajouter()

        resulttrois = compajouter()

        sql ="INSERT INTO composer (idetudiants,idmatieres,idevaluation,note) VALUES(%s,%s,%s,%s)"

        val =(resultprim,resultdeux,resulttrois,notes)

        meconnect.execute(sql, val)

        dernierNom = meconnect.lastrowid

        maBase.commit()

        maBase.close()

        messagebox.showinfo("Information", "Information bien ajouté")

        root.destroy()

        call(["python", "page4.py"])

    except Exception as e:

        messagebox.showinfo("Information", e)

        #retour

        maBase.rollback()

        maBase.close()

def modifier():

    notes = txtnotes.get()

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        resultprim = etuajouter()

        resultdeux = matajouter()

        resulttrois = compajouter()

        sql ="update composer set note=%s where idetudiants= %s and idmatieres= %s and idevaluation= %s"

        val =(notes,resultprim,resultdeux,resulttrois)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "Information modifiée")

        root.destroy()

        call(["python", "page4.py"])

    except Exception as e:

        messagebox.showinfo("Erreur", e)

        #retour

        maBase.rollback()

        maBase.close()

def supprimer():

    maBase = mysql.connector.connect(host="localhost", user="root", password="", database="ecole")

    meconnect = maBase.cursor()

    try:

        resultprim = etuajouter()

        resultdeux = matajouter()

        resulttrois = compajouter()

        sql ="delete from composer where idetudiants= %s and idmatieres= %s and idevaluation= %s"

        val =(resultprim,resultdeux,resulttrois)

        meconnect.execute(sql, val)

        maBase.commit()

        dernierNom = meconnect.lastrowid

        messagebox.showinfo("Information", "Note supprimé")

        root.destroy()

        call(["python", "page4.py"])

    except Exception as e:

        messagebox.showinfo("Erreur", e)

        #retour

        maBase.rollback()

        maBase.close()

def mat(boss):

    Base = mysql.connector.connect(

        host="localhost",

        user="root", #en production autres comptes

        password="", #en production changer de mot de passe

        database="ecole" )#une base en particulier

    meconnect = Base.cursor(buffered=True)

    presql = ("select Nom from etudiants where idetudiants=%s")

    val = (boss,)

    meconnect.execute(presql,val)

    liste\_result = meconnect.fetchall()

    liste\_premier\_result = liste\_result[0]

    result = liste\_premier\_result[0]

    return result

    Base.close()

def matmat(boss):

    Base = mysql.connector.connect(

        host="localhost",

        user="root", #en production autres comptes

        password="", #en production changer de mot de passe

        database="ecole" )#une base en particulier

    meconnect = Base.cursor(buffered=True)

    presql = ("select NomMat from matieres where idmatieres=%s")

    val = (boss,)

    meconnect.execute(presql,val)

    liste\_result = meconnect.fetchall()

    liste\_premier\_result = liste\_result[0]

    result = liste\_premier\_result[0]

    return result

    Base.close()

def mateva(boss):

    Base = mysql.connector.connect(

        host="localhost",

        user="root", #en production autres comptes

        password="", #en production changer de mot de passe

        database="ecole" )#une base en particulier

    meconnect = Base.cursor(buffered=True)

    presql = ("select Nomevaluation from evaluations where idevaluation=%s")

    val = (boss,)

    meconnect.execute(presql,val)

    liste\_result = meconnect.fetchall()

    liste\_premier\_result = liste\_result[0]

    result = liste\_premier\_result[0]

    return result

    Base.close()

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("1375x768+0+0")

root.resizable()

root.configure(background="white")

#Ajouter le titre

lbltitre = Label(root,

    borderwidth = 1,

    relief = GROOVE,

    text = "ENREGISTREMENT DES NOTES",

    font = ("sans serif", 25),

    background = "#FFFFFF",

    fg="#9932CC")

lbltitre.place(x=0,y=0,width= 1375, height=75)

#Nom étudiant

lblnom = Label (root, text="Nom", font=("Arial", 10),bg="#ffffff", fg="#000000")

lblnom.place(x=70, y=160,width=150)

combonom = ttk.Combobox(root,font=("Arial", 9))

combonom['values'] = cmb\_nom()

combonom.place(x=70,y=190,width=300)

#Matiere

lblmat = Label(root,text="Matiere", font=("Arial", 10),bg="#ffffff",fg="#000000")

lblmat.place(x=70, y=240,width=150)

combomat = ttk.Combobox(root,font=("Arial", 9))

combomat['values'] = cmb\_matiere()

combomat.place(x=70,y=270,width=300)

#Composition

lblcomp = Label(root,text="Composition", font=("Arial", 10),bg="#ffffff",fg="#000000")

lblcomp.place(x=70, y=330,width=150)

combocomp = ttk.Combobox(root,font=("Arial", 9))

combocomp['values'] = cmb\_compo()

combocomp.place(x=70,y=360,width=300)

#Notes

lblnotes = Label (root, text="Notes", font=("Arial", 10),bg="#FFFFFF", fg="#000000")

lblnotes.place(x=70, y=410,width=150)

txtnotes = Entry(root, bd=3, font=("Arial", 10))

txtnotes.place(x=70,y=440,width=300)

#AJOUTER

btnajouter = Button(root, text="AJOUTER", font=("Arial", 10),bg="#FFFFFF",fg="#7FFF00", command=ajouter)

btnajouter.place(x=90,y=550,width=85)

#MODIFIER

btnmodifier = Button(root, text="MODIFIER", font=("Arial", 10),bg="#FFFFFF",fg="#9932CC", command=modifier)

btnmodifier.place(x=180,y=550,width=85)

#SUPPRIMER

btnsupprimer = Button(root, text="SUPPRIMER", font=("Arial", 10),bg="#FFFFFF",fg="#A52A2A", command=supprimer)

btnsupprimer.place(x=280,y=550,width=85)

#RESET

# btnreset = Button(root, text="RESET", font=("Arial", 10),bg="#00cc00",fg="white")

# btnreset.place(x=450,y=350,width=85)

#TABLE\_ETUDIANTS

table = ttk.Treeview(root, columns = (1, 2, 3, 4, 5), height = 2, show = "headings")

table.place(x=560,y=150, height=500)

#Entete

table.heading(1, text = "NOM")

table.heading(2, text = "MATIERE")

table.heading(3, text = "EVALUATION")

table.heading(4, text = "NOTES")

#dimentions pour colonnes

table.column(1,width = 150)

table.column(2,width = 150)

table.column(3,width = 150)

table.column(4,width = 50)

maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

meconnect = maBase.cursor()

meconnect.execute("select \* from composer")

plo = meconnect.fetchall()

longueur = len(plo)

for i in range(longueur):

    pla = plo[i]

    long = len(pla)

    for j in range(long):

        if(j==0):

            pli=pla[j]

            toto=mat(pli)

        if(j==1):

            pli=pla[j]

            tete=matmat(pli)

        if(j==2):

            pli=pla[j]

            tyty=mateva(pli)

        if(j==3):

            pli=pla[j]

            note=pli

    table.insert('', END, value = (toto,tete,tyty,note))

maBase.close()

#execution

root.mainloop()

* Page de traitements des bulletins (Page5.py)

#les bibliotheques

import tkinter

from cProfile import label

from tkinter import ttk, Tk

from tkinter import \*

from subprocess import call

from tkinter import messagebox

import mysql.connector

#Fonction pour remplir les valeurs du combo box nom

def cmb\_nom():

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("select Nom from etudiants")

    data = []

    for row in meconnect.fetchall():

        data.append(row[0])

    return data

def cmb\_matiere():

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("select NomMat from matieres")

    data = []

    for row in meconnect.fetchall():

        data.append(row[0])

    return data

def bulletin():

    nom = combonom.get()

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    sql ="SELECT idetudiants from etudiants where Nom=%s"

    val =(nom,)

    meconnect.execute(sql, val)

    maBase.close()

    call(["python", "BDD.py"])

def afficher\_bulletin():

    nom = combonom.get()

    #TABLE\_BULLETIN

    tablbul = ttk.Treeview(root, columns = (1, 2, 3, 4,5), height = 2, show = "headings")

    tablbul.place(x=50,y=250, height=200)

    #Entete

    tablbul.heading(1, text = "MATIERES")

    tablbul.heading(2, text = "DEVOIR")

    tablbul.heading(3, text = "EXAMEN")

    tablbul.heading(4, text = "PROJET")

    tablbul.heading(5, text = "MOYENNE")

    #dimentions pour colonnes

    tablbul.column(1,width = 200)

    tablbul.column(2,width = 100)

    tablbul.column(3,width = 100)

    tablbul.column(4,width = 100)

    tablbul.column(5,width = 100)

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    sql ="SELECT idetudiants from etudiants where Nom=%s"

    val =(nom,)

    meconnect.execute(sql, val)

    liste\_result = meconnect.fetchall()

    liste\_premier\_result = liste\_result[0]

    result = liste\_premier\_result[0]#idetudiants

    maBase.close()

    maBase = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconnect = maBase.cursor()

    meconnect.execute("SELECT \* from evaluations")

    liste\_result = meconnect.fetchall()

    plao=len(liste\_result)

    maBase.close()

    Base = mysql.connector.connect(

    host="localhost",

    user="root", #en production autres comptes

    password="", #en production changer de mot de passe

    database="ecole" )#une base en particulier

    meconect = Base.cursor()

    sqll="SELECT idmatieres, idevaluation, note from composer where idetudiants=%s order by idmatieres ASC,idevaluation ASC"

    vall =(result,)

    meconect.execute(sqll, vall)

    list\_result = meconect.fetchall()

    lon = len(list\_result)

    resulte =""

    donne = [0, 0, 0]

    z=0

    taille=0

    topo=0

    for i in range(lon):

        list\_premier\_result = list\_result[i]

        if resulte!=list\_premier\_result[0]:

            resulte = list\_premier\_result[0]#idmatieres

            Base.close()

            maBase = mysql.connector.connect(

            host="localhost",

            user="root", #en production autres comptes

            password="", #en production changer de mot de passe

            database="ecole" )#une base en particulier

            meconneect = maBase.cursor()

            sqlll="SELECT NomMat from matieres where idmatieres=%s"

            valll =(resulte,)

            meconneect.execute(sqlll, valll)

            list\_resulte = meconneect.fetchall()

            list\_premiere\_result = list\_resulte[0]

            resultee = list\_premiere\_result[0]

            maBase.close()

            moy=0

            for j in range(plao):

                list\_premiere\_result = list\_result[j]

                resulteee = list\_premiere\_result[1]#idevaluation

                maBase = mysql.connector.connect(

                host="localhost",

                user="root", #en production autres comptes

                password="", #en production changer de mot de passe

                database="ecole" )#une base en particulier

                meeconnect = maBase.cursor()

                slq = "SELECT note from composer where idetudiants=%s and idmatieres=%s and idevaluation=%s"

                vla = (result,resulte,resulteee)

                meeconnect.execute(slq, vla)

                answer=meeconnect.fetchall()

                if answer!=[]:

                    donne[j] = answer

                    Donne=donne[j]

                    donnee=Donne[0]

                    donnnee=donnee[0]

                    z+=1

                    if z==2:

                        moy=moy+donnnee\*2

                    else:

                        moy=moy+donnnee

                else:

                    donne[j] = 0

                    z+=1

                maBase.close()

                mog=0

                mog=moy/4

                topo+=mog

            tablbul.insert('', END, value = (resultee,donne[0],donne[1],donne[2],mog))

            taille+=1

    #TABLE\_MOYENNE

    tablnom = ttk.Treeview(root, columns = (1), height = 2, show = "headings")

    tablnom.place(x=650,y=450, height=25)

    tablnom.column(1,width = 200)

    topo=topo/taille

    print(topo)

    tablnom.insert('', END, value = (2,))

    return tablbul,tablnom

#Ma fenetre

root = Tk()

root.title("PROGITUDES")

root.geometry("1375x768+0+0")

root.resizable()

root.configure(background="silver")

#nom

lblnom = Label (root, text="Nom", font=("Arial", 10),bg="#000", fg="white")

lblnom.place(x=70, y=200,width=150)

combonom = ttk.Combobox(root,font=("Arial", 9))

combonom['values'] = cmb\_nom()

combonom.place(x=225,y=200,width=300)

#AJOUt\_grand\_titre

lbltitre = Label(root,

    borderwidth = 1,

    relief = GROOVE,

    text = "BULLETIN DE NOTES",

    font = ("sans serif", 25),

    background = "blue",

    fg="#fffafa")

lbltitre.place(x=0,y=0,width= 1375, height=75)

#MOYENNE\_NOM

lblnom = Label (root, text="Moyenne", font=("Arial", 10),bg="#000", fg="white")

lblnom.place(x=480, y=453,width=150)

#BOUTON\_CALCUL\_GENERER

btnmoyenne = Button(root, text="GENERER", font=("Arial", 10), bg="#00cc00",fg="white",command= afficher\_bulletin)

btnmoyenne.place(x=550, y=200, width=85)

#execution

root.mainloop()