

# Rapport Projet Outils libres en Développement Logiciel

15/12/2020

*Réalisation d'une interface  
graphique : Gestionnaire d'une  
base de données des utilisateurs*



**DB Browser for**  
**SQLite**

TRAN Vincent  
Master 1 Informatique

# **Sommaire**

<b>Présentation du projet.....</b>	<b>3</b>
<b>Création d'une base de données sur DB Browser .....</b>	<b>4</b>
<b>Conception et réalisation d'une interface graphique .....</b>	<b>5</b>
1) Utilisation des librairies.....	5
2) Gérer la connexion sur notre base de données .....	5
3) Ouverture de la fenêtre d'affichage.....	5
<b>Mise en application de l'interface à la base de données .....</b>	<b>7</b>
1) Affichage des données .....	7
2) Ajout des données .....	8
3) Suppression des données choisi dans l'ID .....	9
4) Mise à jour des données choisi par l'ID.....	10
5) Fermeture de la fenêtre principale .....	11
<b>conclusion .....</b>	<b>11</b>

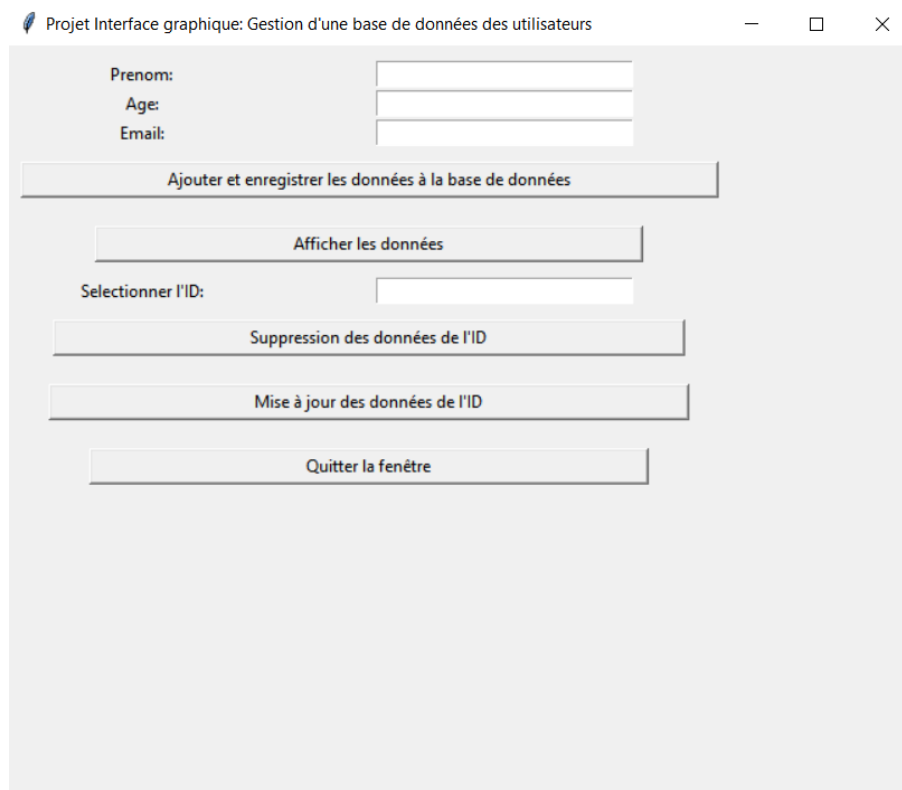
## Présentation du projet

J'ai réalisé une interface graphique pour concevoir un gestionnaire d'une base de données des utilisateurs programmé en Python et en SQL dans le logiciel Visual Studio Code et avec DB Browser pour créer un fichier d'une base de données en format .db. Cette l'interface graphique est utilisée par le package Tkinter et sur cette fenêtre il est composé de 5 fonctionnalités pour gérer une base de données :

- L'ajout et l'enregistrement des données
- L'affichage des données
- La suppression des données
- La mise à jour des données
- Quitter la fenêtre

L'objectif de ce projet consiste à manipuler une base de données dans une interface graphique pour comprendre les différentes bases élémentaires d'une base de données des utilisateurs.

Voici la conception de l'interface graphique dans lequel j'avais compilé mon script :



Nous allons l'utilisation de DB Browser pour créer d'une base de données ensuite, je vais expliquer la conception et la réalisation de l'interface graphique que j'avais utilisé pour l'ouverture de la fenêtre et des fonctions spécifiques pour manipuler les données. Puis, je vais expliquer les différentes étapes de tests à utiliser pour mettre en application l'interface de cette base de données. Enfin, je vais conclure l'avis de ce projet ce que j'ai bien appris.

## Création d'une base de données sur DB Browser

Sur le logiciel DB Brower, j'ai créé une table de données codé en SQL sous le nom utilisateurs avec 3 noms de variables : *Name*, *Age*, *Email*. Ces trois variables vont servir pour importer sur le programme en Python pour pouvoir manipuler notre gestion de base de données directement sur l'interface graphique.

```
CREATE TABLE `utilisateurs` (  
  `Name` varchar(255) DEFAULT NULL,  
  `Age` int(32) DEFAULT NULL,  
  `Email` varchar(255) DEFAULT NULL  
)
```

Nom de la variable	Type	Taille	Valeur nul
<b>Name</b>	Chaine de caractère	255 octets	Oui
<b>Age</b>	Entier	32 octets	Oui
<b>Email</b>	Chaine de caractère	255 octets	Oui

Au niveau les données, j'ai défini 6 lignes de données dans le tableau de données :

	Name	Age	Email
	Filtre	Filtre	Filtre
1	Jean-Paul	26	Jean-Paul@gmail.com
2	Alexandre	20	alexandre@gmail.com
3	Christophe	24	Christophe@gmail.com
4	Laura	18	Laura@hotmail.fr
5	Lucy	23	Lucy@free.fr
6	Julien	21	julien@yahoo.com

En connaissant notre table de données, nous pourrons désormais importer cette base de données pour pouvoir manipuler sur notre interface graphique codé en Python et en SQL.

## Conception et réalisation d'une interface graphique

### 1) Utilisation des librairies

Au niveau des librairies, j'ai utilisé bien évidemment la librairie Tkinter qui est un module intégré à la bibliothèque standard de Python qui offre des moyens pour créer des interfaces graphiques via Python. Il est disponible sur Windows et la plupart des systèmes d'Unix. Ensuite le SQLite est une librairie qui fournit une base de données mais qui ne nécessite pas de processus serveur et permet d'accéder à la base de données du langage de requête SQL qu'on va importer notre fichier database.db.



```
from tkinter import *  
import sqlite3
```

### 2) Gérer la connexion sur notre base de données

La connexion de la base de données doit être comprise pendant toute la durée de vie et toutes les commandes sont exécutées dans le contexte de la session de base de données du fichier database.db en utilisant grâce à la méthode cursor().

```
conn = sqlite3.connect('C:\\Users\\vince\\Projet_Logiciel\\database.db')  
c = conn.cursor()
```

Il est important pour chaque fonction spécifique lorsqu'on va coder du SQL d'accéder toujours au fichier database.db pour pouvoir se connecter directement à notre base de données de l'utilisateur.

```
#Chargement de données  
conn.commit()  
  
#Fermeture de la connexion  
conn.close()
```

A la fin de notre programme principal, il est important de valider sa transaction en SQL avec la méthode commit() et d'utiliser close() pour enregistrer notre base de données.

### 3) Ouverture de la fenêtre d'affichage

On doit tout d'abord d'appeler la classe Tk() pour créer notre première fenêtre d'affichage et j'ai défini une dimension de **650 X 550** avec le titre :

**« Projet Interface graphique : Gestion d'une base de données des utilisateurs »**

```
root = Tk()  
root.title("Projet Interface graphique: Gestion d'une base de données des utilisateurs")  
root.geometry("650x550")
```

A la fin du programme principal, nous allons utiliser la méthode mainloop() pour pouvoir ouvrir notre première fenêtre lorsqu'on a compilé le programme.

```
#Ouverture de la fenêtre principale  
root.mainloop()
```

#### 4) Utilisation des fonctionnalités dans la fenêtre

Je vais créer 4 zones de textes (**représentés en bleus**) qui sont le nom, l'âge, l'email pour permettre de rentrer les valeurs sur l'interface graphique et le numéro de l'ID pour déterminer à quelle ligne d'une base de données je vais supprimer ou modifier toutes les données. Et notamment 4 étiquettes (**représentés en rouge**) pour afficher le texte sur l'interface graphique.

```
#Creation des zones de textes  
name = Entry(root,width=30)  
name.grid(row=0,column=1,padx=20,pady=(10,0))  
  
age = Entry(root,width=30)  
age.grid(row=1,column=1)  
  
email = Entry(root,width=30)  
email.grid(row=2,column=1)  
  
delete_box = Entry(root,width=30)  
delete_box.grid(row=9,column=1)
```

```
#Creation des étiquettes attribués aux zones de textes  
name_label = Label(root,text="Prenom:")  
name_label.grid(row=0,column=0,pady=(10,0))  
  
age_label = Label(root,text="Age:")  
age_label.grid(row=1,column=0)  
  
email_label = Label(root,text="Email:")  
email_label.grid(row=2,column=0)  
  
ID_label = Label(root,text="Selectionner l'ID:")  
ID_label.grid(row=9,column=0)
```

Et ensuite j'ai créé 5 boutons (représentés en verts qui vont permettre d'exécuter à chaque fonctionnalité qui sont :

- L'ajout des données
- L'affichage des données
- La suppression des données choisi dans l'ID
- La mise à jour des données choisi dans l'ID
- La fermeture de la fenêtre

```
#Creation Bouton d'ajout de données  
submit_btn = Button(root, text="Ajouter et enregistrer les données à la base de données",command=submit)  
submit_btn.grid(row=5,column=0,columnspan=2,pady=10,padx=10,ipadx=100)  
  
#Creation Bouton d'affichage de données  
query_btn = Button(root,text="Afficher les données",command=query)  
query_btn.grid(row=7,column=0,columnspan=2,pady=10,padx=10,ipadx=137)  
  
#Creation Bouton de suppression de données  
delete_btn = Button(root,text="Suppression des données de l'ID",command=delete)  
delete_btn.grid(row=10,column=0,columnspan=2,pady=10,padx=10,ipadx=136)  
  
#Creation Bouton Mise à jour de données  
edit_btn = Button(root,text="Mise à jour des données de l'ID",command=edit)  
edit_btn.grid(row=12,column=0,columnspan=2,pady=10,padx=10,ipadx=142)  
  
#Creation Bouton Quitter la page  
quit_btn = Button(root,text="Quitter la fenêtre",command=leave)  
quit_btn.grid(row=13,column=0,columnspan=2,pady=10,padx=10,ipadx=150)
```

Projet Interface graphique: Gestion d'une base de données des utilisateurs

Prenom:  
Age:  
Email:

Ajouter et enregistrer les données à la base de données

Afficher les données

Selectionner l'ID:

Suppression des données de l'ID

Mise à jour des données de l'ID

Quitter la fenêtre

## Mise en application de l'interface à la base de données

### 1) Affichage des données

Lors du premier affichage des données sur notre base de données database.db, nous avons bien reçu les données pour la liste des prénoms, des âges et des emails qui sont affichés entre le bouton afficher les données et la zone de texte pour sélectionner l'ID :

Projet Interface graphique: Gestion d'une base de données des utilisateurs

Prenom:  
Age:  
Email:

Ajouter et enregistrer les données à la base de données

Afficher les données

Jean-Paul	26	Jean-Paul@gmail.com	1
Alexandre	20	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6

Selectionner l'ID:

Suppression des données de l'ID

Mise à jour des données de l'ID

Quitter la fenêtre

## 2) Ajout des données

Pour ajouter des données, on doit taper les valeurs les trois premières zones de textes (Prénom, Age et Email) et il faudra bien respecter les types de données si non la base de données n'envoie pas de requêtes.

Si on souhaite ajouter le prénom : **Vincent**, l'âge : **21** et l'email : **vincenttran94@hotmail.fr** et qu'on clique le bouton sur Ajouter et enregistrer les données à la base de données, et puis d'afficher les données, les entrées de données reçoivent bien la requête qui sont bien pris en compte sur ma base de données.

Prenom:	<input type="text" value="Vincent"/>
Age:	<input type="text" value="21"/>
Email:	<input type="text" value="vincenttran94@hotmail.fr"/>
<input type="button" value="Ajouter et enregistrer les données à la base de données"/>	

Les données seront automatiquement supprimées sur les zones de textes et le message a dit que : « *Vos données ont été tout ajoutés* »

Prenom:	<input type="text"/>
Age:	<input type="text"/>
Email:	<input type="text"/>
<input type="button" value="Ajouter et enregistrer les données à la base de données"/>	
Vos données ont été tout ajoutées	

Jean-Paul	26	Jean-Paul@gmail.com	1
Alexandre	20	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6
Vincent	21	vincenttran94@hotmail.fr	7



Si on souhaite maintenant de ne pas d'ajouter les données, sur les zones de textes, on aperçoit un autre message que : « Vos données n'ont pas été tout ajoutés » mais après tout automatiquement il ajoute le numéro 8 de l'ID sans de mettre les données.

Prenom:

Age:

Email:

Ajouter et enregistrer les données à la base de données

Vos données n'ont pas été tout ajoutées

Jean-Paul	26	Jean-Paul@gmail.com	1
Alexandre	20	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6
Vincent	21	vincenttran94@hotmail.fr	7
			8

### 3) Suppression des données choisi dans l'ID

Pour supprimer des données, il faudra obligatoirement choisir le numéro de l'ID intégré dans la base de données, si on tape sur la zone de texte le numéro 1 de l'ID et qu'on appuie sur le bouton de suppression des données, on aperçoit que toutes les données dans l'ID numéro 1 ont été tout supprimés.

Selectionner l'ID:

Vos données ont été bien supprimés

Afficher les données

Alexandre	20	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6
Vincent	21	vincenttran94@hotmail.fr	7
lol	2	dqdq	8

Selectionner l'ID:

Suppression des données de l'ID

En revanche, si on ne tape pas de numéro de l'ID, un message s'affichera que :

**Vos données n'ont été pas été supprimés**

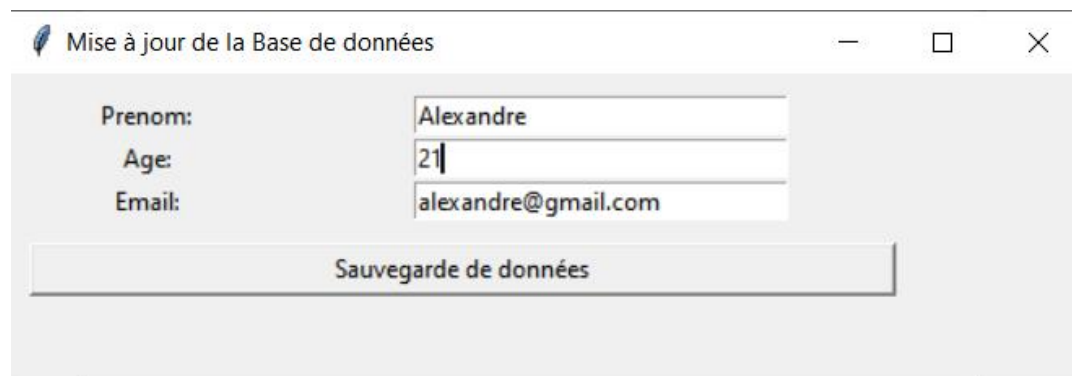
Et donc, toutes les autres lignes de données qui n'ont pas été choisis par l'ID sont bien conservés.

#### 4) Mise à jour des données choisi par l'ID

Si on souhaite de mettre à jour les données choisies par un numéro de l'ID, une nouvelle fenêtre apparaîtra et mettra automatiquement les données contenues sur notre base de données et donc la requête a été bien appliqué.

Si on entre le numéro de l'ID 2, et qu'on clique sur « Mise à jour des données de l'ID », on aperçoit une nouvelle fenêtre avec des données contenus dans les zones de textes et donc on pourra modifier directement sur la zone de texte. Je modifie par exemple l'âge de Alexandre 21 ans à place de 20 ans et une fois modifié, on pourra sauvegarder sur notre de base de données et notre requête SQL procède un enregistrement. Et la deuxième fenêtre de l'interface fermera et envoie un message dans la fenêtre principale : « Vos données ont été bien à jour »

#### Ouverture de la deuxième fenêtre



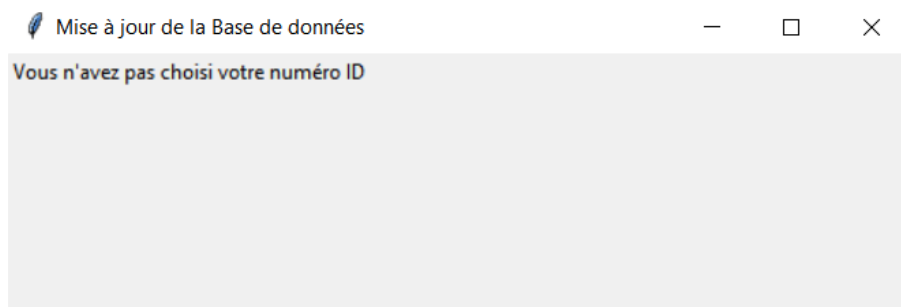
#### **Avant la mise à jour**

Alexandre	20	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6
Vincent	21	vincenttran94@hotmail.fr	7

#### **Après la mise à jour**

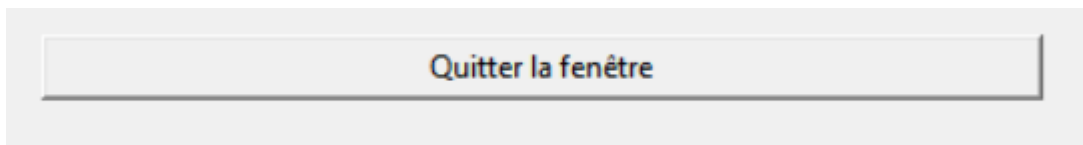
Alexandre	21	alexandre@gmail.com	2
Christophe	24	Christophe@gmail.com	3
Laura	18	Laura@hotmail.fr	4
Lucy	23	Lucy@free.fr	5
Julien	21	julien@yahoo.com	6
Vincent	21	vincenttran94@hotmail.fr	7

Remarque: Si on souhaite de ne pas mettre un numéro d'ID, un message d'erreur apparaîtra sur la deuxième fenêtre : et donc il faudra quitter sur la croix en haut à droite pour quitter la fenêtre.



### 5) Fermeture de la fenêtre principale

On peut désormais fermer notre fenêtre principale en cliquant sur le bouton : Quitter la fenêtre pour pouvoir fermer l'exécution du programme et d'enregistrer tout les données contenus sur notre base de donnée : database.db



## Conclusion

Ce projet était un projet très intense car j'ai énormément fait des recherches à ce qui concerne le choix et la manipulation de l'interface graphique et notamment aussi comment marche les différentes procédures de tests de base de données pour apprendre le codage SQL. Et il y avait plein de projets à réaliser pour cette année en Master.

Grâce à ce projet, j'ai appris pour la première fois d'utiliser un gestionnaire d'une base de données des utilisateurs avec une interface graphique programmé en Python.

Le projet n'est pas lié d'un projet professionnel mais me permet surtout de comprendre les différentes bases d'une gestion de base de données intégré sur une interface graphique.