

```

# -*- coding: utf-8 -*-
"""
Created on Fri Jan 5 19:47:00 2024

@author: DELL
"""

import tkinter as tk
import mysql.connector
from connexion_db import connect_to_db

database = connect_to_db()
def clear_entries(*entries):
    for entry in entries:
        entry.delete(0, tk.END)

def ajouter_produit(database):

    def add_product():
        nom_produit = entry_nom_produit.get()
        description = entry_description.get()
        quantite = entry_quantite.get()
        prix_achat = entry_prix_achat.get()
        prix_vente = entry_prix_vente.get()

        try:
            cursor = database.cursor()
            cursor.execute("SELECT MAX(id_produit) + 1 FROM produit")
            next_id = cursor.fetchone()[0]
            query = "INSERT INTO produit (id_produit, nom_produit, description, quantite, prix_achat, prix_vente) VALUES (%s, %s, %s, %s, %s, %s)"
            cursor.execute(query, (next_id, nom_produit, description, quantite, prix_achat, prix_vente))
            database.commit()
            print("Produit ajouté avec succès.")
            clear_entries(entry_nom_produit, entry_description, entry_quantite, entry_prix_achat, entry_prix_vente)
        except mysql.connector.Error as err:
            print(f"Erreur lors de l'ajout du produit : {err}")

    root = tk.Tk()
    root.title("Ajout de Produit")
    root.configure(bg='#7D4FFE')

    label_nom = tk.Label(root, text="Nom du produit:", bg='#7D4FFE', fg="white")
    label_nom.pack()

    entry_nom_produit = tk.Entry(root)
    entry_nom_produit.pack()

    label_description = tk.Label(root, text="Description:", bg='#7D4FFE', fg="white")
    label_description.pack()

    entry_description = tk.Entry(root)
    entry_description.pack()

    label_quantite = tk.Label(root, text="Quantité:", bg='#7D4FFE', fg="white")
    label_quantite.pack()

    entry_quantite = tk.Entry(root)
    entry_quantite.pack()

    label_prix_achat = tk.Label(root, text="Prix d'achat:", bg='#7D4FFE', fg="white")
    label_prix_achat.pack()

```

```

entry_prix_achat = tk.Entry(root)
entry_prix_achat.pack()

label_prix_vente = tk.Label(root, text="Prix de vente:", bg='#7D4FFE', fg="white")
label_prix_vente.pack()

entry_prix_vente = tk.Entry(root)
entry_prix_vente.pack()

btn_ajouter = tk.Button(root, text="Ajouter", command=add_product)
btn_ajouter.pack()

root.mainloop()

```

```

def lister_produits(database):
    try:
        cursor = database.cursor()
        cursor.execute("SELECT CONCAT('id: ', id_produit, ', nom: ', nom_produit, ', desc: ', desc) AS produit")
        result = cursor.fetchall()
        root = tk.Tk()
        root.title("Liste des Produits")

        text_area = tk.Text(root, height=10, width=80)
        text_area.pack()

        for row in result:
            text_area.insert(tk.END, row[0] + "\n")

        root.mainloop()
    except mysql.connector.Error as err:
        print(f"Erreur lors de la récupération des produits : {err}")

```

```

def chercher_produit(database):
    def rechercher():
        produit_recherche = entry_produit.get()
        try:
            cursor = database.cursor()
            query = f"SELECT CONCAT('id: ', id_produit, ', nom: ', nom_produit, ', desc: ', desc) AS produit WHERE nom_produit LIKE '{produit_recherche}%"
            cursor.execute(query)
            result = cursor.fetchall()

            text_area.delete(1.0, tk.END)

            for row in result:
                text_area.insert(tk.END, row[0] + "\n")
        except mysql.connector.Error as err:
            print(f"Erreur lors de la recherche de produit : {err}")

    root = tk.Tk()
    root.title("Recherche de Produit")
    root.configure(bg='#7D4FFE')

    label = tk.Label(root, text="Nom du Produit à rechercher:")
    label.pack()

    entry_produit = tk.Entry(root)
    entry_produit.pack()

    search_button = tk.Button(root, text="Rechercher", command=rechercher)

```

```

search_button.pack()

text_area = tk.Text(root, height=10, width=80)
text_area.pack()

root.mainloop()

def supprimer_produit(database):
    def delete_product():
        product_name = entry_nom_produit.get()
        try:
            cursor = database.cursor()
            query = "DELETE FROM produit WHERE nom_produit = %s"
            cursor.execute(query, (product_name,))
            database.commit()
            print("Produit supprimé avec succès.")
            clear_entries(entry_nom_produit)
        except mysql.connector.Error as err:
            print(f"Erreur lors de la suppression du produit : {err}")

    root = tk.Tk()
    root.title("Suppression de Produit")

    label_nom = tk.Label(root, text="Nom du produit à supprimer:")
    label_nom.pack()

    entry_nom_produit = tk.Entry(root)
    entry_nom_produit.pack()

    btn_supprimer = tk.Button(root, text="Supprimer", command=delete_product)
    btn_supprimer.pack()

    root.mainloop()

def get_products(database):
    try:
        cursor = database.cursor()
        cursor.execute("SELECT id_produit, nom_produit, quantite FROM produit")
        return cursor.fetchall()
    except mysql.connector.Error as err:
        print(f"Erreur lors de la récupération des produits : {err}")
        return []

def update_quantity(database, product_id, new_quantity):
    try:
        cursor = database.cursor()
        query = "UPDATE produit SET quantite = %s WHERE id_produit = %s;"
        cursor.execute(query, (new_quantity, product_id))
        database.commit()
    except mysql.connector.Error as err:
        print(f"Erreur lors de la mise à jour de la quantité : {err}")

def inventaire(database):
    def update():
        for product_id, entry in entries.items():
            new_quantity = entry.get()
            update_quantity(database, product_id, int(new_quantity))
        print("Produits mise à jour avec succès !")

    root = tk.Tk()
    root.title("Inventaire")

```

```

products = get_products(database)

rows = []
entries = {}

for i, product in enumerate(products):
    id_produit, nom_produit, quantite = product
    row = []

    label_id = tk.Label(root, text=f"ID: {id_produit}")
    label_id.grid(row=i, column=0)
    row.append(label_id)

    label_nom = tk.Label(root, text=f"Nom: {nom_produit}")
    label_nom.grid(row=i, column=1)
    row.append(label_nom)

    label_quantite = tk.Label(root, text="Quantité:")
    label_quantite.grid(row=i, column=2)
    row.append(label_quantite)

    entry_quantite = tk.Entry(root)
    entry_quantite.insert(tk.END, quantite)
    entry_quantite.grid(row=i, column=3)
    row.append(entry_quantite)

    rows.append(row)
    entries[id_produit] = entry_quantite

update_button = tk.Button(root, text="Mettre à jour", command=update)
update_button.grid(row=len(products), columnspan=4)

root.mainloop()

def modifier_produit(database):
    def rechercher_produit():
        nom_produit = entry_nom_produit.get()
        try:
            cursor = database.cursor()
            query = "SELECT * FROM produit WHERE nom_produit = %s"
            cursor.execute(query, (nom_produit,))
            result = cursor.fetchone()

            if result:
                root_modif = tk.Tk()
                root_modif.title("Modifier Produit")

                labels = ["Nom du Produit:", "Description:", "Quantité:", "Prix Achat:", "Prix \
                entries = []
                for i, label_text in enumerate(labels):
                    label = tk.Label(root_modif, text=label_text)
                    label.grid(row=i, column=0)
                    entry = tk.Entry(root_modif)
                    entry.insert(tk.END, result[i+1])
                    entry.grid(row=i, column=1)
                    entries.append(entry)

                def modifier():
                    new_values = [entry.get() for entry in entries]
                    update_product(database, result[0], new_values)

```

```

        clear_entries(*entries)
        clear_entries(entry_nom_produit)
        btn_modifier = tk.Button(root_modif, text="Modifier", command=modifier)
        btn_modifier.grid(row=len(labels), columnspan=2)

        root_modif.mainloop()
    else:
        print("Produit non trouvé.")
    except mysql.connector.Error as err:
        print(f"Erreur lors de la recherche du produit : {err}")
def update_product(database, product_id, new_values):
    try:
        cursor = database.cursor()
        query = "UPDATE produit SET nom_produit = %s, description = %s, quantite = %s, prix_"
        cursor.execute(query, (*new_values, product_id))
        database.commit()
        print("Produit mis à jour avec succès !")
    except mysql.connector.Error as err:
        print(f"Erreur lors de la mise à jour du produit : {err}")

root = tk.Tk()
root.title("Modifier Produit")

label_nom_produit = tk.Label(root, text="Nom du Produit à Modifier:")
label_nom_produit.pack()

entry_nom_produit = tk.Entry(root)
entry_nom_produit.pack()

btn_rechercher = tk.Button(root, text="Rechercher", command=rechercher_produit)
btn_rechercher.pack()

root.mainloop()

```