import sqlite3

from datetime import datetime

# Create a SQLite database and tasks table if it doesn't exist

def create\_database():

conn = sqlite3.connect('tasks.db')

cursor = conn.cursor()

cursor.execute('''

CREATE TABLE IF NOT EXISTS tasks (

id INTEGER PRIMARY KEY,

task TEXT NOT NULL,

priority TEXT,

due\_date DATE,

completed BOOLEAN

)

''')

conn.commit()

conn.close()

# Add a task to the database

def add\_task(conn, task, priority, due\_date):

cursor = conn.cursor()

cursor.execute('''

INSERT INTO tasks (task, priority, due\_date, completed)

VALUES (?, ?, ?, ?)

''', (task, priority, due\_date, False))

conn.commit()

# Remove a task from the database

def remove\_task(conn, task\_id):

cursor = conn.cursor()

cursor.execute('DELETE FROM tasks WHERE id = ?', (task\_id,))

conn.commit()

# Mark a task as completed in the database

def complete\_task(conn, task\_id):

cursor = conn.cursor()

cursor.execute('UPDATE tasks SET completed = 1 WHERE id = ?', (task\_id,))

conn.commit()

# List all tasks from the database

def list\_tasks(conn):

cursor = conn.cursor()

cursor.execute('SELECT \* FROM tasks')

tasks = cursor.fetchall()

if not tasks:

print("No tasks in the list.")

else:

print("Tasks:")

for task in tasks:

task\_id, task\_text, priority, due\_date, completed = task

status = "Completed" if completed else "Not Completed"

due\_date\_str = due\_date.strftime("%Y-%m-%d") if due\_date else "No due date"

print(f"ID: {task\_id}, Priority: {priority}, Due Date: {due\_date\_str}, Status: {status}, Task: {task\_text}")

# Main function

def main():

create\_database()

while True:

conn = sqlite3.connect('tasks.db')

print("\nOptions:")

print("1. Add a task")

print("2. Remove a task")

print("3. Mark a task as completed")

print("4. List tasks")

print("5. Quit")

choice = input("Enter your choice (1/2/3/4/5): ")

if choice == "1":

task = input("Enter the task to add: ")

priority = input("Enter the priority (high/medium/low): ").lower()

due\_date\_str = input("Enter the due date (YYYY-MM-DD, leave empty for no due date): ")

due\_date = datetime.strptime(due\_date\_str, "%Y-%m-%d") if due\_date\_str else None

add\_task(conn, task, priority, due\_date)

elif choice == "2":

task\_id = int(input("Enter the task ID to remove: "))

remove\_task(conn, task\_id)

elif choice == "3":

task\_id = int(input("Enter the task ID to mark as completed: "))

complete\_task(conn, task\_id)

elif choice == "4":

list\_tasks(conn)

elif choice == "5":

conn.close()

print("Goodbye!")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()