Complete mini project on CRUD application using SQLite and Python.

Create, Read, Update, and Delete for employees' database

Bellow are the SQLite queries and python used in this project

```
import sqlite3

# Create database connection
conn = sqlite3.connect('employee.db')

# Create a cursor
c = conn.cursor()

# Create a table
#c.execute('CREATE TABLE IF NOT EXISTS employees (name TEXT,value INTEGER)')

#Insert records into the table
many_employees = [
        ('Abul', 1357),
        ('Adolfo', 1224),
        ('Alexander', 2296),
        ('Amber', 1145),
        ('Amy', 4359),
        ('Andy', 1966),
        ('Anna', 4040),
        ('Antony', 449),
        ('Ashley', 8151),
        ('Borja', 9428),
        ('Cecilia', 2136),
        ('Christopher', 9035),
        ('Dan', 1475),
        ('Dario', 284),
        ('David', 948),
        ('Elike', 1860),
        ('Ella', 4549),
        ('Ellie', 5736),
        ('Elliot', 1020),
        ('Emily', 7658),
        ('Faye', 7399),
        ('Fern', 1422),
```

```
('Francisco', 5028),
 ('Frank', 3281),
 ('Gary', 9190),
('Germaine', 6437),
 ('Greg', 5929),
('Harvey', 8471),
 ('Helen', 963),
('Huzairi', 9491),
 ('Izmi', 8324),
 ('James', 6994),
('Jarek', 6581),
  ('Jim', 202),
 ('John', 261),
 ('Jose', 1605),
('Josef', 3714),
('Karthik', 4828),
('Katrin', 5393),
  ('Lee', 269),
 ('Luke', 5926),
('Madiha', 2329),
 ('Marc', 3651),
('Marina', 6903),
 ('Mark', 3368),
('Marzena', 7515),
('Mohamed', 1080),
('Nichole', 1221),
('Nikita', 8520),
('Oliver', 2868),
('Patryk', 1418),
 ('Paul', 4332),
('Ralph', 1581),
('Raymond', 7393),
('Roman', 4056),
 ('Ryan', 252),
 ('Sara', 2618),
 ('Sean', 691),
 ('Seb', 5395),
('Sergey', 8282),
('Shaheen', 3721),
('Sharni', 7737),
 ('Sinu', 3349),
('Stephen', 8105),
```

```
                                        ('Tim', 8386),
                                        ('Tina', 5133),
                                        ('Tom', 7553),
                                        ('Tony', 4432),
                                        ('Tracy', 1771),
                                        ('Tristan', 2030),
                                        ('Victor', 1046),
                                        ('Yury', 1854),
                                    ]

c.executemany(" INSERT INTO employees VALUES (?,?)", many_employees)

                        #Add a record to the table
                        def add_item(name, value):
                    conn = sqlite3.connect('employee.db')
                            c = conn.cursor()
        c.execute("INSERT INTO employees VALUES (?,?)", (name, value))

                                conn.commit()
                                conn.close()

                    #Add many records to the table
                        def add_items(lists):
         c.executemany(" INSERT INTO employees VALUES (?,?)", (lists))
               print("Records are been added succesfully....")

                                conn.commit()
                                conn.close()

              #Query the database & returnd all the records
                            def show_all():
                         conn = sqlite3.connect('employee.db')
                                c = conn.cursor()
                         c.execute("SELECT * FROM employees")
                                items = c.fetchall()

                    print("    A LIST OF THE EMPLOYEES    " )
                   print(" -----**------**------**------- " )
                   print("EMPLOYEE NAME" + "\t|EMP NUMBER " )
                   print("-------------" + "\t|---------- " )

                                for item in items:
```

```
                    print(item[0] + " \t\t| "  + str(item[1]))

                                conn.commit()
                                conn.close()

                            #delete a record from table
                                def delete_item(id):
                        c.execute("DELETE from employees WHERE rowid = (?)", id)
                          print("The record is been deleted successfully....")
                                conn.commit()
                                conn.close()


                            #Add many records to the table
                                def add_items(lists):
                        c.executemany(" INSERT INTO employees VALUES (?,?)", (lists))
                              print("Records are been added successfully....")

                                conn.commit()
                                conn.close()

## Increment the value by 1 where the field Name begins with 'E'           ##
## and by 10 where Name begins with 'G'                                    ##
## and all others by 100                                                   ##
                                        #
                                   ##  UPDATE employees
                                     #  SET value += 1
                                     #  WHERE LIKE 'e%'
                                        #
                                   ##  UPDATE employees
                                  #  SET value = value + 10
                                     #  WHERE LIKE 'e%'
                                        #
                                   ##  UPDATE employees
                                 #  SET value = value + 100
                                   #  WHERE LIKE '[!eg]%'

## List the sum of all Values for all names that begin with A,B or C but  ##
## only present the data where the summed values are greater than or       ##
## equal to 11171                                                          ##
                                        #
                                  #    SELECT name
```

```
                    #   FROM employees
 #   WHERE name Like'[a-c]%' IN (SELECT name, SUM(value)
      #                              FROM employees
      #                               GROUP BY name
#                                HAVING SUM(value)>= 11171)
                               #
                               #




                          conn.commit()
                          conn.close()
```