Додаток А.1

**Код програми-драйвера для бази даних (файл engine.py)**

import os

import random

import time

import sqlite3

import hashlib

from random import choice

from string import ascii\_letters

BASE\_DIR = os.path.dirname(os.path.abspath(\_\_file\_\_))

class DataBase:

def \_\_init\_\_(self):

self.name\_db = 'db\_for\_php'

self.\_\_connect\_to\_db(self.name\_db)

try:

self.\_\_create\_table\_in\_db()

except:

pass

self.\_\_close\_db()

def \_\_connect\_to\_db(self, db\_name):

self.conn = sqlite3.connect(db\_name)

self.cursor\_db = self.conn.cursor()

def \_\_create\_table\_in\_db(self):

sql\_stmt = '''CREATE TABLE operators (id INTEGER PRIMARY KEY AUTOINCREMENT, operator VARCHAR)'''

self.cursor\_db.execute(sql\_stmt)

sql\_stmt = '''CREATE TABLE example (id INTEGER PRIMARY KEY AUTOINCREMENT, text VARCHAR)'''

self.cursor\_db.execute(sql\_stmt)

sql\_stmt = '''CREATE TABLE description (id INTEGER PRIMARY KEY AUTOINCREMENT, text VARCHAR)'''

self.cursor\_db.execute(sql\_stmt)

def \_\_close\_db(self):

self.cursor\_db.close()

self.conn.close()

def \_add\_new\_operator(self, operator):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''INSERT INTO main.operators (operator)

VALUES (?)'''

self.cursor\_db.execute(sql\_stmt, (operator,))

self.conn.commit()

self.\_\_close\_db()

def \_add\_example(self, text):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''INSERT INTO example (text)

VALUES (?)'''

self.cursor\_db.execute(sql\_stmt, (text,))

self.conn.commit()

self.\_\_close\_db()

def \_add\_description(self, text):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''INSERT INTO description (text)

VALUES (?)'''

Додаток А.2

self.cursor\_db.execute(sql\_stmt, (text,))

self.conn.commit()

self.\_\_close\_db()

def add\_article(self, operator, text\_for\_example, text\_for\_description):

try:

self.\_add\_new\_operator(operator)

self.\_add\_example(text\_for\_example)

self.\_add\_description(text\_for\_description)

except:

pass

else:

return 0

def get\_all\_articles(self):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''SELECT oper.id, oper.operator, ex.text, descr.text

FROM main.operators oper, main.example ex, main.description descr

WHERE oper.id = ex.id AND oper.id = descr.id'''

self.cursor\_db.execute(sql\_stmt)

return self.cursor\_db.fetchall()

def get\_article(self, id):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''SELECT oper.id, oper.operator, ex.text, descr.text

FROM main.operators oper, main.example ex, main.description descr

WHERE oper.id = (?) AND ex.id = oper.id AND descr.id = oper.id'''

self.cursor\_db.execute(sql\_stmt, (id,))

self.conn.commit()

self.\_\_close\_db()

def edit\_article(self, id, operator, text\_for\_example, text\_for\_description):

try:

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''UPDATE main.operators

SET operator = ?

WHERE id = ?'''

self.cursor\_db.execute(sql\_stmt, (operator, id))

sql\_stmt = '''UPDATE main.example

SET text = ?

WHERE id = ?'''

self.cursor\_db.execute(sql\_stmt, (text\_for\_example, id))

sql\_stmt = '''UPDATE main.description

SET text = ?

WHERE id = ?'''

self.cursor\_db.execute(sql\_stmt, (text\_for\_description, id))

self.conn.commit()

self.\_\_close\_db()

except:

pass

else:

return 0

def delete\_article(self, id):

Додаток А.3

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''DELETE FROM main.operators

WHERE main.operators.id = ?'''

self.cursor\_db.execute(sql\_stmt, (id,))

self.conn.commit()

sql\_stmt = '''DELETE FROM main.example

WHERE main.example.id = ?'''

self.cursor\_db.execute(sql\_stmt, (id,))

self.conn.commit()

sql\_stmt = '''DELETE FROM main.description

WHERE main.description.id = ?'''

self.cursor\_db.execute(sql\_stmt, (id,))

self.conn.commit()

self.\_\_close\_db()

def search\_operators(self, query):

self.\_\_connect\_to\_db(self.name\_db)

sql\_stmt = '''SELECT oper.id, oper.operator, ex.text, descr.text

FROM main.operators oper, main.example ex, main.description descr

WHERE (oper.operator LIKE "%{0}%" OR ex.text LIKE "%{1}%" OR descr.text LIKE "%{2}%")

AND oper.id = ex.id AND ex.id = descr.id'''.format(query, query, query)

self.cursor\_db.execute(sql\_stmt)

return self.cursor\_db.fetchall()