

**Keywords:** *Python, calculator, programming, variables, functions, exception handling, code, comments.*

**Науковий керівник: Тищенко С.І.,**  
*канд. пед., наук, доцент, доцент кафедри економічної кібернетики,  
комп'ютерних наук та інформаційних технологій,  
Миколаївський національний аграрний університет  
м. Миколаїв, Україна*

УДК 004.62

## **РОБОТА З БАЗАМИ ДАНИХ В PYTHON: SQL ТА NOSQL**

**Тимошенко Єлизавета Сергіївна,**  
*здобувач вищої освіти спеціальності 122 «Комп'ютерні науки»  
Миколаївський національний аграрний університет  
м. Миколаїв, Україна*

**Анотація:** *В дослідженні розглянуто бази даних SQL та NoSQL і робота з ними в Python. Наведено приклад використання модуля sqlite3, що дозволяє працювати з локальними SQL базами даних без зовнішніх залежностей. Розглянуто можливість використання бібліотеки pymongo для роботи з MongoDB в Python. Також проаналізовано особливості баз даних SQL та NoSQL і сфери їх використання.*

**Ключові слова:** *SQL, NoSQL, модуль sqlite3, бібліотека pymongo.*

SQL база даних (Structured Query Language) – це колекція даних, яка організована та зберігається за допомогою системи управління базами даних (СУБД), що підтримує мову SQL для маніпулювання цими даними. Вона складається з таблиць, де дані представлені у вигляді рядків (записів) та стовпців (атрибутів). Широко використовується в різних сферах, таких як бізнес, наука, освіта та інформаційні технології, завдяки її здатності зберігати та оброблювати великі обсяги даних та забезпечувати ефективний доступ до неї.

Python має бібліотеку SQLite, яка дозволяє працювати з локальними SQL базами даних без зовнішніх залежностей, через вбудований модуль sqlite3. Для підключення до віддалених SQL баз даних, таких як MySQL або PostgreSQL, можна використовувати сторонні бібліотеки, такі як mysql-connector-python та psycopg2. Також популярним вибором є бібліотека SQLAlchemy, яка пропонує ORM (Object Relational Mapping) для більш високорівневої роботи з базами даних.

Приклад використання модуля sqlite3:

```

import sqlite3
# Створення підключення до бази даних (або її створення, якщо вона не існує)
conn = sqlite3.connect('example.db')
# Створення курсору
cursor = conn.cursor()
# Створення таблиці
cursor.execute("CREATE TABLE IF NOT EXISTS stocks (date text, trans text,
symbol text, qty real, price real)")
# Вставка даних
cursor.execute("INSERT INTO stocks VALUES ('2022-01-05','BUY','RHAT',100,35.14)")
# Збереження (commit) змін
conn.commit()
# Закриття з'єднання
conn.close()

```

NoSQL база даних (Not Only SQL) – це тип бази даних, який відрізняється від традиційних SQL баз даних у використанні неструктурованих або півструктурованих даних та у відмові від фіксованої схеми таблиць. NoSQL бази даних – це не реляційні бази даних, що можуть зберігати дані у форматах, таких як key-value, document, wide-column, або graph.

NoSQL бази даних були розроблені для роботи з великими обсягами даних, розподіленими середніми та високими навантаженнями. Деякі приклади NoSQL баз даних включають MongoDB, Cassandra, Redis, Couchbase, Amazon DynamoDB та інші. Вони широко використовуються в різних галузях, таких як веб-розробка, аналітика даних, інтернет речей (IoT) та інші, де вимагається обробка великих обсягів неструктурованих даних або потрібне гнучке масштабування.

MongoDB – популярна NoSQL база даних, яка використовує документоорієнтований підхід. Для роботи з MongoDB в Python можна використовувати бібліотеку pymongo. Для початку роботи необхідно встановити pymongo та підключитися до сервера MongoDB:

```

pip install pymongo
from pymongo import MongoClient
# Створення підключення до MongoDB
client = MongoClient('mongodb://localhost:27017/')
# Вибір бази даних
db = client['test-database']
# Вибір колекції
collection = db['test-collection']
# Вставка документу
post = {"author": "Mike", "text": "My first blog post!", "tags": ["mongodb",
"python", "pymongo"]}
post_id = collection.insert_one(post).inserted_id

```

Під час вибору бази даних слід враховувати потреби вашого проекту.

SQL (реляційні бази даних) використовують таблиці зі структурованими даними, організованими у рядки та стовпці. Вони використовують схему, що передбачає фіксовані типи даних та обов'язковість вказівки структури даних. NoSQL (нереляційні бази даних) дозволяють зберігати неструктуровані або півструктуровані дані. Вони можуть використовувати різні моделі даних, такі як документи, графи, ключ-значення, стовпчасті тощо, залежно від конкретного випадку застосування.

SQL зазвичай складно горизонтально масштабувати, тобто розділити дані між декількома серверами для збільшення продуктивності та доступності, натомість NoSQL часто підтримують горизонтальне масштабування, що дозволяє розподіляти дані на кілька серверів.

SQL використовує SQL (Structured Query Language) для запитів до даних. Може бути потрібно виконання складних JOIN-операцій для об'єднання даних з різних таблиць, а в NoSQL може бути менша складність у формулюванні запитів, але це може залежати від конкретного типу NoSQL бази даних та її моделі даних.

SQL вимагає строгої схеми даних, де структура таблиць та типи даних фіксовані, а NoSQL дозволяє гнучкішу схему даних, де можна зберігати дані з різними структурами без необхідності відтворення схеми на всіх рівнях.

Враховуючи особливості кожної з баз, можна сформулювати сфери їх застосування. SQL зазвичай використовується для більш традиційних застосувань, таких як електронна комерція, фінанси, управління клієнтами тощо, а NoSQL застосовується там, де потрібно обробляти великі обсяги даних або працювати з неструктурованими даними, такими як аналітика великих даних, рекомендаційні системи, IoT (Internet of Things) тощо.

В одному проєкті можливе використання і SQL, і NoSQL баз даних, залежно від потреб. Інтеграція обох типів баз у Python проєкт залежить від специфічних вимог. Можна мати окремі модулі або класи для взаємодії з кожним типом бази даних і використовувати їх у коді відповідно до потреб обробки даних.

### Список використаних джерел:

1. <https://mate.academy/blog/python/sql-nosql-databases-python/>
2. <https://uk.wikipedia.org/wiki/SQL>
3. <https://alternativescience.net/programming/242-sql-chy-nosql-os-v-chomu-pytannya/>

***Annotation:** The research examines SQL and NoSQL databases and working with them in Python. An example of using the sqlite3 module is provided, which allows you to work with local SQL databases without external dependencies. The possibility of using the pymongo library to work with MongoDB in Python is considered. The peculiarities of SQL and NoSQL databases and their areas of use are also analyzed.*

***Keywords:** SQL, NoSQL, sqlite3 module, pymongo library.*

**Науковий керівник: Пархоменко О.Ю.,**  
к.ф.-м.н., доцент,