

УДК 004.8

ОСОБЛИВОСТІ ВИКОРИСТАННЯ РІЗНИХ СУБД У PYTHON-ПРОЄКТАХ

Запорожцев Д.С., Запорожцев М.С.

Львівський національний університет природокористування

Використання різних СУБД в ІТ-проєктах обумовлене передусім направленістю та ресурсоемністю додатків, які розробляються. Сучасні рішення пропонують багато варіантів СУБД. Мета даної роботи - дослідити особливості впровадження та використання баз даних для проєкту, який написано мовою програмування Python. В якості прикладів СУБД були обрані MySQL та Oracle.

Як відомо, база даних (БД) - організована структура, яка переважно призначена для зберігання та обробки інформації, частіше всього, великого обсягу даних. [1-2]

В сучасних системах БД активно використовуються в таких галузях, як електронна комерція (управління замовленнями, каталогами товарів, кошиками покупців), медицина (зберігання електронних медичних записів, історій хвороби) фінансова (облік транзакцій, управління клієнтськими рахунками), аналітика (обробка великих обсягів даних для досліджень і прогнозів).

Без БД неможливо уявити створення комерційних інтернет додатків, таких як інтернет-магазини.

В них можуть бути представлені як продукції схожої тематики, так і зовсім різної, але всі дані про неї (кількість на складі, кількість замовлень даної , термін придатності, тощо) зберігається в єдиній БД, що забезпечує швидкий доступ до них, її оновлення, видалення або обробку. Крім того, БД допомагають уникнути дублювання даних, забезпечуючи їх цілісність і узгодженість.

Python, як одна з найпопулярніших мов програмування, забезпечує широкий інструментарій для роботи з БД різних типів, включаючи вбудовані бібліотеки (sqlite3) і сторонні рішення для взаємодії з популярними системами управління баз даних (СУБД), зокрема - SQLite, MySQL, PostgreSQL, MongoDB, Redis, Oracle, тощо. [3]

Існує кілька основних типів БД, які використовуються залежно від завдання. Зазвичай, виділяють реляційні БД (іноді їх називають SQL-базами) та нереляційні (або NoSQL) системи. Слід зауважити, що Python є потужним інструментом для роботи з обома варіантами баз даних, але все ж таки більш популярними є реляційні бази даних, через їх широкий функціонал, підтримку та просту реалізацію.

Реляційні бази даних (RDBMS) - найбільш поширений тип БД, що зберігає інформацію у вигляді таблиць, де кожен рядок відповідає запису, а стовпчики — полям із даними. Для взаємодії з такими базами використовується мова SQL (Structured Query Language).

MySQL – відома реляційна СУБД із відкритим вихідним кодом, популярна завдяки своїй стабільності та продуктивності. [4] Python пропонує кілька бібліотек для інтеграції з MySQL, таких як mysql-connector-python, PyMySQL та MySQLdb. MySQL добре підходить для веб-додатків середнього розміру і великих проєктів, які потребують надійного зберігання великих обсягів даних.

Oracle – високопродуктивна реляційна СУБД, орієнтована на корпоративний сегмент. [5] Python-додатки можуть працювати з Oracle через бібліотеку cx_Oracle. Ця БД є стандартом для великих компаній завдяки своїй надійності, продуктивності і широким можливостям захисту даних.

Результати порівняння цих двох СУБД наведені в табл.1.

З цього порівняння можемо зробити висновок, що остаточний вибір бази даних для проєкту залежить, перш за все, від потреб та вимог, які поставлені замовником: обсягу даних, складності запитів та вимог до продуктивності.

Таблиця 1 - Порівняння СУБД MySQL та Oracle

Характеристики	MySQL	Oracle
Тип бази даних	Реляційна база даних з відкритим вихідним кодом.	Реляційна база даних корпоративного рівня.
Призначення	Зазвичай використовується для веб-додатків, середніх і великих проєктів.	Призначена для великих корпоративних систем із високими вимогами до безпеки та продуктивності.
Області застосування	Веб-додатки.	ERP, CRM, банківські системи, облік великих даних.
Продуктивність	Висока швидкість обробки запитів, оптимальна для читання і запису даних середнього обсягу.	Висока продуктивність для великих обсягів даних, підтримка складних обчислень і паралельної обробки.
Розширюваність	Легке горизонтальне масштабування, наприклад, через реплікацію або шардинг.	Підтримка вертикального масштабування, складних кластерів і функцій.
Сховище даних	Підтримка різних механізмів зберігання (InnoDB, MyISAM).	Оптимізовано для роботи з великими обсягами даних і підтримує Data Warehousing.
Підтримка транзакцій	Підтримує ACID через механізм InnoDB, але менше налаштованих опцій.	Повна підтримка ACID з розширеними можливостями конфігурації транзакцій.

Обидві розглянуті СУБД мають свої особливості, переваги та обмеження.

Використання Python у цій сфері забезпечує гнучкість та ефективність у розробці додатків будь-якого масштабу.

Література:

1. Що таке бази даних та їх роль у програмуванні на Python. URL: <https://foxminded.ua/baza-danykh-python/>
2. Що таке база даних? URL: <https://apeps.kpi.ua/shco-take-basa-danykh>.
3. Python and MySQL: The Complete Practical Guide, Part 1. URL: <https://robotdreams.cc/blog/484-python-ta-mysql-povniy-praktichniy-posibnik-ch-1>
4. MySQL 8.4 Reference Manual :: 1.2.1 What is MySQL? URL: <https://dev.mysql.com/doc/refman/8.4/en/what-is-mysql.html>
5. Application development with Oracle Database. URL: <https://www.oracle.com/cis/database/technologies/application-development.html>.