

**Олег Вікторович Резніченко<sup>1</sup>, Оксана Анатоліївна Ляшенко<sup>2</sup>,  
Вікторія Вікторівна Архипова<sup>3</sup>**

<sup>1</sup> аспірант кафедри інформаційних систем

Український державний університет науки і технологій (Дніпро, Україна)

**E-mail:** [rov250977@gmail.com](mailto:rov250977@gmail.com). **ORCID:** <https://orcid.org/0009-0001-0194-5079>

<sup>2</sup> кандидат технічних наук, доцент кафедри інформаційних систем

Український державний університет науки і технологій (Дніпро, Україна)

**E-mail:** [liashenko\\_o\\_a@ukr.net](mailto:liashenko_o_a@ukr.net). **ORCID:** <https://orcid.org/0000-0002-9983-5504>

<sup>3</sup> кандидат технічних наук, старший науковий співробітник наукового центра інноваційних матеріалів та технологій

Український державний університет науки і технологій (Дніпро, Україна),

доцент кафедри цивільної інженерії, технологій будівництва і захисту довкілля

Дніпровський державний аграрно-економічний університет (Дніпро, Україна)

**E-mail:** [arh.v.1006@gmail.com](mailto:arh.v.1006@gmail.com). **ORCID:** <https://orcid.org/0000-0002-9722-5622>

### **ВИБІР МІЖ РЕЛЯЦІЙНИМИ ТА НЕРЕЛЯЦІЙНИМИ БАЗАМИ ДАНИХ**

*Стаття є оглядово-інформаційною. У цьому дослідженні розглянуто основні характеристики сучасних моделей баз даних, що використовуються для зберігання та обробки інформації, яка може бути отримана в будь-якій галузі науки, суспільства та промисловості. Розглянуто ключові характеристики та особливості сховищ даних. Визначено, що існує дві основні моделі баз даних – реляційні та нереляційні. Сформульовано їхні недоліки та переваги, що обмежують або, навпаки, розширюють їхні функціональні можливості. Зроблено висновки щодо перспектив використання сховищ даних.*

**Ключові слова:** інформація; реляційні бази даних; нереляційні бази даних; NoSQL; SQL.

*Рис.: 2. Бібл.: 25.*

**Актуальність теми дослідження.** Інформація відіграє значну роль у функціонуванні будь-яких галузей життя людини. Невелику її кількість можна зберігати на одному комп'ютері або сервері, але в теперішній час дані досягли таких обсягів, що неможливо їх розташувати таким чином. Це робить необхідним створювати та вдосконалювати технології і методи розміщення й обробки інформації та узгодженої роботи багатьох машин, що можуть знаходитися просторово на великій відстані.

**Постановка проблеми.** При розробці систем збереження інформації необхідно враховувати, що сумісне функціонування комп'ютерів може бути описано за допомогою теорему Брюера – досягти узгодженості, доступності і стійкості збереженої інформації водночас неможливо, що призводить до виникнення низки проблем, які необхідно вирішити. Так, до питань, що підлягають розв'язанню, належить:

- перевищення об'ємів інформації над продуктивністю пристрою, що її обробляє;
- недостатня швидкість роботи з даними (запису, оновлення, переміщення);
- різноманітність інформації та складність або неможливість її класифікації та упорядкування;
- наявність великої кількості даних, які не є корисними.

Тому підхід до вирішення цих питань не може однозначно бути вирішеним і повинен бути комплексним.

**Аналіз останніх досліджень і публікацій.** Для зберігання і обробки інформації використовуються бази даних.

Існує декілька типів баз даних залежно від їхніх властивостей: мережеві та локальні, розподілені і нерозподілені, зосереджені і розосереджені, репліковані і нерепліковані, агрегатні і безагрегатні, структуровані і неструктуровані, об'єктноорієнтовані, реляційні і нереляційні, SQL та NoSQL та інші.

Вважається, що найбільш поширеними є реляційні та нереляційні бази даних (що також називаються SQL та NoSQL відповідно).

**Виділення недосліджених частин загальної проблеми.** Проведений аналіз попередніх досліджень показав, що створення баз даних та їхня швидка й надійна робота можлива тільки після вивчення та урахування всіх їхніх можливостей і обмежень.

**Метою роботи** є огляд сучасного стану організації зберігання і обробки даних, їх використання в різних галузях виробництва та формування подальших досліджень.

**Виклад основного матеріалу.** SQL бази даних отримали свою назву від мови запитів, що допомагає створювати цей тип сховищ даних і обробляти існуючу інформацію (робити нові записи, отримувати дані, редагувати інформацію шляхом видалення та оновлення, виконувати запити) та управляти даними за допомогою систем управління реляційною базою даних (Relational Database Management System - RDBMS) [1; 2].

Виник цей тип баз даних в 70-ті роки минулого століття, ставши основою інформаційних систем. Засновником вважається Е.Ф. Кодд, відповідно до уявлень якого пропонується організувати дані в таблиці (зв'язки), де кожна таблиця є набором рядків (кортежів) і стовпців (атрибутів), це спрощує роботу з даними та їх пошук [3]. Подальше використання цієї моделі бази даних такими компаніями, як Oracle, IBM і Microsoft призвело до її поширення [4, 5].

На сьогодні найбільш популярними типами баз даних SQL є такі (рис. 1).

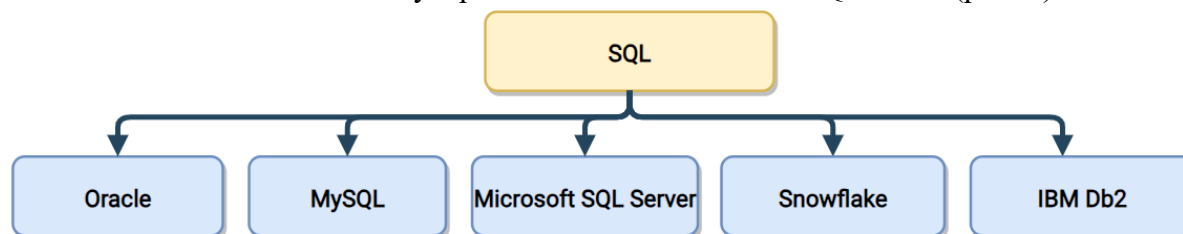


Рис. 1. Найбільш поширені типи баз даних SQL

Джерело: розроблено авторами.

Реляційні бази даних, удосконалюючись протягом багатьох років, для більшості компаній стали стандартом проєктування, управління та збереження інформації [6; 7].

SQL бази даних мають такі переваги [8]:

- використовують порівняно просту та зрозумілу організацію даних;
- мають визначений стандарт, що дозволяє керувати значною кількістю інформації без додаткового написання програмних кодів;
- використовують математичну логіку, засновану на теорії множин та дискретній математиці;
- застосовують структуровані й організовані дані, що забезпечує більшу ефективність роботи;
- є продуктивними, оскільки дозволяють за допомогою запитів отримувати велику кількість записів, що знаходяться в різних таблицях;
- є стабільними та передбачуваними, адже відповідає принципам ACID (атомарність, узгодженість, ізоляція та довговічність);
- підтримують широкий набір програмних інструментів та сумісні з великою кількістю комп'ютерних програм.

Незважаючи на велику кількість позитивних факторів, SQL бази даних мають і недоліки [9-11]:

- використовується для роботи тільки зі структурованими даними і не припустимі для обробки даних іншого типу;
- мають обмеження по гнучкості структури даних, оскільки структура даних і таблиць повинна бути визначена на початку проєктування, а подальші її зміни ускладнені;

- мають обмеження по масштабованості, адже при збільшенні обсягів даних різко зменшується швидкість обробки інформації;
- можуть містити надмірну інформацію, оскільки при її обробці створюються додаткові таблиці;
- несуть високі операційні витрати на збереження інформації;
- мають складність роботи з ієрархічними структурами.

Ці недоліки, що призводили до обмеження або неможливості використання SQL баз даних, стали причиною виникнення баз даних іншого типу – нереляційних, або NoSQL [11-12]. Відповідно до назви, бази даних цього виду взагалі не використовують мову запитів SQL, хоча за іншими даними, найменування може означати і «не тільки SQL». Цей тип баз даних є нетабличною формою представлення інформації, тому перед її створенням немає потреби визначати схему бази даних.

Можна сказати, що NoSQL тип баз даних [4] є схожим на папку з файлами, що може зберігати файли будь-якого типу.

Виділяють декілька типів подібних сховищ інформації, які відрізняються за метою використання, структурою [12-13] та джерелом створення.

Взагалі, всі NoSQL бази даних можна поділити на групи:

- модель ключ-значення (Key /Value);
- таблична / колонкоорієнтована модель (Tabular Column Oriented);
- документоорієнтована модель (Document Oriented);
- графова модель (Graph).

Різними компаніями створено багато видів баз даних кожної групи (рис. 2) [14-15].

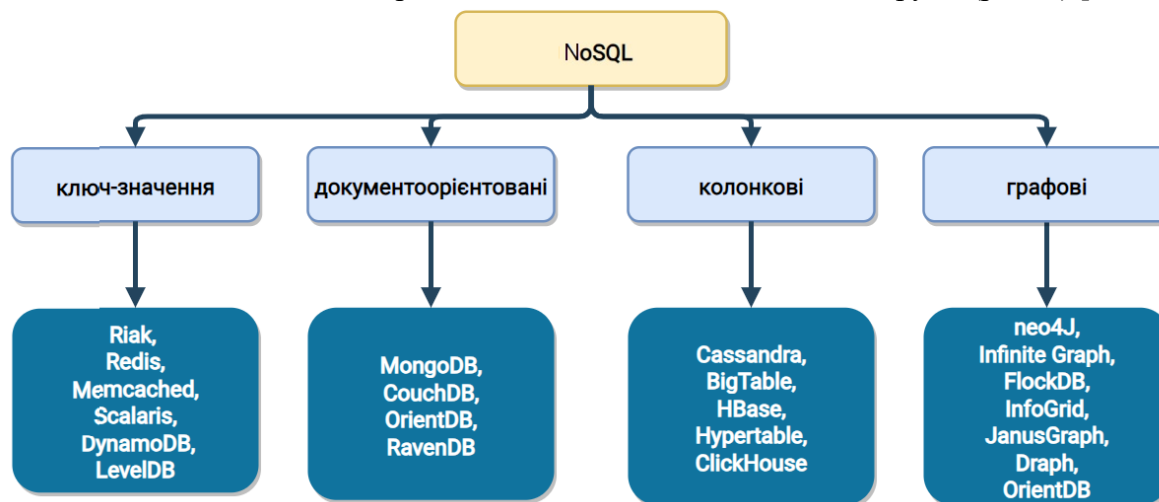


Рис. 2. Найбільш поширені види баз даних NoSQL

Джерело: розроблено авторами

На відміну від реляційних баз даних, нереляційні:

- не зберігають інформацію у вигляді таблиць;
- використовують гнучкі схеми, що дає можливість пришвидшити розробку та забезпечити поетапну реалізацію;
- не використовують мову запитів SQL;
- підтримують можливість горизонтального масштабування і роботи з даними big data [16-19];
- дають можливість управляти даними в реальному часі;
- підтримують використання агрегатів і кластерів;
- застосовують динамічні гнучкі схеми [20].

Отже, вважаючи на різноманітність ознак нереляційних баз даних, не існує чіткого визначення поняття NoSQL баз даних [21].

Перевагами баз даних NoSQL є такі [22-24]:

- використовуються для зберігання великих обсягів інформації;
- дозволяють здійснювати паралельну обробку інформації на розподілених серверах;
- підтримують горизонтальне масштабування;
- забезпечують високу продуктивність завдяки безсхемній моделі;
- дають можливість управляти даними у реальному часі;
- підтримують використання агрегатів і кластерів;
- забезпечують можливість зберігати неструктуровані або напівструктуровані дані;
- мають економічність у контексті їх розгортання при збільшенні обсягів даних і транзакцій.

Звичайно, ці бази даних мають і недоліки, пов'язані з:

- обмеженістю складних запитів;
- відсутністю стандартизації;
- недостатньою стабільністю.

Однак ці недоліки компенсуються роботою з великими об'ємами даних та масштабованістю, тому використання NoSQL баз даних значно поширюється.

Кожен із типів NoSQL баз даних має свої особливості, переваги і обмеження. Основними функціональними властивостями баз даних можна означити узгодженість даних, транзакції, властивості запитів, структуру даних та масштабування.

*Ключ-значення.* Є найбільш простою моделлю сховища інформації, оскільки представляє собою таблицю з двох стовпців, один з яких є ключем, а другий – його значенням. Доступ до даної бази даних відбувається за первинним ключем, що різко збільшує продуктивність бази та можливість до масштабування [17; 20; 22; 24].

Властивість стосується операцій над окремим ключем, оскільки ці операції можуть отримувати, записувати та видаляти дані за окремим ключем.

Для баз даних цього типу характерна підсумкова узгодженість даних: або залишити новий запис та прибрати старий, або залишити дві, спитавши про це користувача. Транзакції реалізуються частіше за усе за допомогою забезпечення концепції кворуму. Запити виконуються за ключем, і, як правило, неможливо провести обробку за значенням. Допускається масштабування за допомогою фрагментації, причому значення ключа буде визначати його місцезнаходження.

Бази даних ключ-значення краще не використовувати для випадків, якщо між різними наборами даних необхідно встановити кореляцію; якщо необхідно виконувати запит за значеннями; якщо необхідно обробляти одразу декілька ключів.

Більш підходящими випадками використання баз даних ключ-значення є зберігання профілів користувачів, даних із сайтів та вебдодатків та ін.

*Документні бази даних* [24; 25]. Складаються з окремих елементів (документів), що можуть представляти собою файли, масиви та інші об'єкти; ідентифікація відбувається за ключем, що створюється самою базою даних. Узгодженість даних забезпечується за допомогою наборів реплік, причому задаючи їх кількість, можна або посилити узгодженість, або продуктивність. Транзакції є атомарними на рівні окремого документа, і неможливі, якщо містять більше однієї операції. Документні бази даних забезпечують різні можливості для запитів: вони можуть бути матеріалізовані та динамічні, «відображення-згортка» та ін. Масштабування відбувається шляхом додавання допоміжних вузлів (горизонтальне), причому немає потреби в перезавантаженні системи; або шляхом фрагментації даних у тому ж вузлу.

Документні бази даних не слід вживати для складних транзакцій. Використовуються вони для реєстрації різних подій різного типу; даних аналізу реального часу; інформаційних додатків для вебсайтів.

Стовпцеві бази даних представляють собою множину стовпців, асоційованих з ключем рядку [24]. Основною одиницею зберігання є стовпець, що складається з пари «ім'я-значення» з вказівкою часу. Передбачається узгодженість запису та читання, щоб забезпечувати високу доступність з компромісом у відношенні затримки збереження даних. Транзакції є атомарними, оскільки дані або будуть записані, або ні. Колонкові бази даних мають високу доступність, оскільки в них усі вузли в кластері мають однакові права. Запити відрізняються залежно від типу бази даних, або являють собою набір команд, або спеціальну мову запитів. Масштабування відбувається додаванням додаткових вузлів.

Бази даних використовуються для реєстрації подій, зберігання записів блогів, аналітичних показників. Неєфективно використовувати для транзакційних систем.

Графові бази даних є систематичним набором даних, які дозволяють зберігати сутності (вузли) та відносини між ними (ребра). Більшість графових баз даних не підтримує розподілу вузлів по серверах, але якщо база даних працює на кластері, дані є узгодженими, і запис на ведучий вузол синхронізується з веденими, що доступні для читання (тобто доступність забезпечується реплікацією веденими вузлами). Запити забезпечуються в багатьох випадках спеціальною мовою запитів. Масштабування відбувається або шляхом додавання додаткових ведених вузлів для читання даних, запис проводити на ведучих вузлах; або шляхом фрагментації даних. Графові бази ефективно застосувати для областей з багатьма взаємними зв'язками (наприклад, соціальних мереж), і не підходить для випадків, якщо є необхідність змінювати властивості в усіх вузлах.

Є ще декілька типів NoSQL баз даних, але вони менш поширені.

**Висновки.** Протягом тривалого часу SQL бази даних широко використовувались у світі. Але виникнення багатьох складних питань, пов'язаних із необхідністю обробки запитів вебдодатків, збільшенням кількостей даних та користувачів, узгодженням даних різних типів, обробкою величезних обсягів геопросторових даних та ін. призвело до необхідності створення баз даних нового типу. Використання NoSQL спростило розробку завдяки гнучкій безструктурній схемі, дозволило обробку неструктурованої інформації та масштабування даних.

### Список використаних джерел

1. Gruber, M. *Understanding SQL* / M. Gruber. – San Francisco : Sybex, 1990. – 434 p.
2. Mpinda S. A. T. *From Relational Database to Column-Oriented NoSQL Database: Migration Process* / S. A. T. Mpinda, P. A. Bungama, L. G. Maschietto // *International Journal of Engineering Research and Technology*. – 2015. – Vol. 4, № 5. – Pp. 399-403.
3. Codd, E. F. *A relational model of data for large shared data banks* / E. F. Codd // *Communications of the ACM*. – 1970. – Vol. 13, № 6. – Pp. 377–387.
4. Babucea A.-G. *SQL or NoSQL Databases? Critical differences* / A.-G. Babucea // *Annals of the „Constantin Brâncuși” University of Târgu Jiu. Economy Series*. – 2021. – № 1. – Pp. 53-59.
5. Резніченко, В. А. *60 років базам даних* / В. А. Резніченко // *Проблеми програмування*. – 2021. – № 3. – С. 40-71.
6. ISO/IEC 9075-1:2023, *Information technology – Database languages – SQL Part 1: Framework (SQL/Framework)* [Electronic resource]. – Access mode: <https://www.iso.org/ru/standard/76583.html>.
7. Severance, C. Elizabeth Fong: *Creating the SQL Database Standards* / C. Severance // *Computer*. – 2014. – Vol. 47, № 8. – Pp. 7-8.
8. Beaulieu, A. *Learning SQL: Generate, Manipulate, and Retrieve Data* / A. Beaulieu. – O'Reilly Media, 2020. – 380 p.

9. Коломійцев, О. В. Аналіз переваг і недоліків реляційних та нереляційних баз даних логістичних інформаційних систем / О. В. Коломійцев [та ін.] // Society and Science: Interconnection : Proceedings of the 1st international scientific and practical conference, [Porto], 26-28 жовтня 2022 р. – Porto, 2022. – С. 416-422.
10. Ali W. Comparison between SQL and nosql databases and their relationship with big data analytics / W. Ali [et al.] // Asian Journal of Research in Computer Science. – 2019. – Vol. 4, № 2. – Pp. 1-10.
11. Stonebraker, M. SQL databases v. NoSQL databases / M. Stonebraker // Communications of the ACM. – 2010. – Vol. 53, №. 4. – Pp. 10-11.
12. Bjeladinovic, S. A fresh approach for hybrid SQL/NoSQL database design based on data structuredness / S. Bjeladinovic // Enterprise Information Systems. – 2018. – Vol. 12, №. 8-9. – P. 1202–1220.
13. Hnatushenko, V. V. Non-relational approach to developing knowledge bases of expert system prototype / V. V. Hnatushenko, Vik. V. Hnatushenko, N. L. Dorosh, N. O. Solodka, O. A. Liashenko // Naukovyi visnyk natsionalnoho hirnychoho universytetu. – 2022. – № 2. – Pp. 112–117.
14. Ляшенко, О. А. Сравнительный анализ выполнения запросов к серверам баз данных MYSQL и MONGODB / О. А. Ляшенко, О. О. Конашков, Н. О. Солodka // Вісник Херсонського національного технічного університету. – 2019. – № 4. – С. 114-124.
15. Головка, А. О. Проектування застосування щодо підбору житла з використанням реляційної та нереляційної баз даних / А. О. Головка, О. А. Ляшенко // Комп'ютерне моделювання та оптимізація складних систем (КМОСС-2023): матеріали VIII Міжнар. науково-техн. конф., (Дніпро, 1–3 листоп. 2023 р.), ДВНЗ УДХТУ. – Дніпро, 2023. – С. 157-158.
16. Rodriguez, J. Distributed Architecture of Mobile GIS Application Using NoSQL Database / Jonathan Rodriguez [et al.] // International Journal of Information and Electronics Engineering. – 2017. – Vol. 7, № 6. – Pp. 156-160.
17. Krechowicz, A. Highly Scalable Distributed Architecture for NoSQL Datastore Supporting Strong Consistency / A. Krechowicz, S. Deniziak, G. Lukawski // IEEE Access. – 2021. – Vol. 9. – Pp. 69027-69043.
18. Ali, A. A State of Art Survey for Big Data Processing and NoSQL Database Architecture / A. Ali [et al.] // International Journal of Computing and Digital Systems. – 2023. – Vol. 14, № 1. – Pp. 297-309.
19. Shamsi, J. A. Big Data Systems: A 360-Degree Approach / J. A. Shamsi, M. A. Khojaye. – Taylor & Francis Group, 2021. – 312 p.
20. Rahman, R. Scalable Security Analytics Framework Using NoSQL Database / R. ur Rahman, D. S. Tomar // International Journal of Database. Theory and Application. – 2017. – Vol. 10, № 11. – Pp. 27–46.
21. Sattar, A. Incorporating NoSQL into a database course / A. Sattar, T. Lorenzen, K. Nallamaddi // ACM Inroads. – 2013. – Vol. 4, № 2. – Pp. 50-53.
22. Waldman, D. A. The acceptability of 360 degree appraisals: A customer-supplier relationship perspective / D. A. Waldman, D. E. Bowen // Human Resource Management. – 1998. – Vol. 37, № 2. – Pp. 117-129.
23. Abdullah, A. Leveraging Advanced Machine Learning Techniques for Enhanced Intrusion and Fraud Detection in NoSQL Database Systems / A. Abdullah, T. Arjunan // International Journal of Applied Machine Learning and Computational Intelligence. – 2023. – Vol. 13, №. 11.
24. Sadalage, P. NoSQL distilled: A brief guide to the emerging world of polyglot persistence / P. Sadalage, M. Fowler. – Upper Saddle River, NJ : Addison-Wesley, 2013. – 188 p.
25. Ляшенко, О. А. Анализ производительности баз данных PostgreSQL/POSTGIS и MONGODB для геопространственных запросов / О. А. Ляшенко, С. Н. Литвинов, Н. А. Солodka // Вісник Кременчуцького національного університету імені Михайла Остроградського. – 2019. – № 6. – С. 60–67.

## References

1. Gruber, M. (1990). *Understanding SQL*. Sybex.
2. Mpinda, S. A. T., Bungama, P.A. & Maschietto, L. G. (2015). From relational database to column-oriented nosql database: Migration process. *International Journal of Engineering Research and Technology*, 2015. – Vol. 4, no. 05. – Pp. 399-403.
3. Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6), 377–387. <https://doi.org/10.1145/362384.362685>.

4. Babucea, A.-G. (2021). SQL or NoSQL Databases? Critical differences. *Annals of the „Constantin Brâncuși” University of Târgu Jiu. Economy Series, 1*, 53-59.
5. Reznichenko, V. A. (2021). 60 rokov bazam danykh [60 years of databases]. *Problemy prohranuvannia – Programming problems, 3*, 40–71. <http://doi.org/10.15407/pp2021.03.040>.
6. International Organization for Standardization. (2023). *Information technology – Database languages – SQL Part 1: Framework (SQL/Framework) – Requirements (ISO/IEC 9075-1:2023)*.
7. Severance, C. (2014). Elizabeth Fong: Creating the SQL database standards. *Computer, 47*(8), 7–8. <https://doi.org/10.1109/mc.2014.209>.
8. Beaulieu, A. (2020). *Learning SQL: Generate, manipulate, and retrieve data*. O'Reilly Media.
9. Kolomiitsev, O.V., Startsev, V.V., Tretiak, V.F., Nikorchuk, A.I., Shapovalov, O.I., Poltavskiy, E.M., Chernenko, P.V., Kryvchun, V.I., Zakharchenko, V.V., Rybalchenko, A.O. (2022). Analiz perevah i nedolikhv reliatsiinykh ta nereliatsiinykh baz danykh lohistrychnykh informatsiinykh system [Analysis of advantages and disadvantages of realistic and unrealistic databases of logistics information systems]. *Proceedings of the 1st international scientific and practical conference: Society and Science: Interconnection* (pp. 416–422). Scientific Publishing Center “InterConf”. <https://archive.interconf.center/index.php/conference-proceeding/issue/view/26-28.11.2022/144>.
10. Ali, W., Shafique, M. U., Majeed, M. A., & Raza, A. (2019). Comparison between SQL and nosql databases and their relationship with big data analytics. *Asian Journal of Research in Computer Science, 4*(2), 1–10. <https://doi.org/10.9734/ajrcos/2019/v4i230108>.
11. Stonebraker, M. (2010). SQL databases v. NoSQL databases. *Communications of the ACM, 53*(4), 10–11. <https://doi.org/10.1145/1721654.1721659>.
12. Bjeladinovic, S. (2018). A fresh approach for hybrid SQL/NoSQL database design based on data structuredness. *Enterprise Information Systems, 12*(8-9), 1202–1220. <https://doi.org/10.1080/17517575.2018.1446102>.
13. Hnatushenko, V. V., Hnatushenko, Vik. V., Dorosh, N. L., Solodka, N. O., & Liashenko, O. A. (2022). Non-relational approach to developing knowledge bases of expert system prototype. *Naukovyi visnyk natsionalnoho hirnychoho universytetu, 2*(188), 112–117. <https://doi.org/10.33271/nvngu/2022-2/112>.
14. Liashenko, O. A., Konashkov, O. O., & Solodka, N. O. (2019). Sravnitel'nyi analiz vipolneniya zaprosiv k serveram baz dannikh MYSQL i MONGODB [Comparative analysis of request execution to MYSQL and MONGODB database servers]. *Visnyk Khersonskoho natsionalnoho tekhnichnoho universytetu – Herald of the Kherson National Technical University, 4*(71), 114–124. <https://doi.org/10.35546/kntu2078-4481.2019.4.13>.
15. Holovko, A.O. (2023). Proiektuvannia zastosunku shchodo pidboru zhytla z vykorystanniam reliatsiinoi ta nereliatsiinoi baz danykh. [Designing an application for housing selection using real and non-real databases], [Computer modelling and optimisation of complex systems], A.O. Holovko, O. A. Liashenko. *Kompiuterne modeliuvannia ta optymizatsiia skladnykh system (KMOSS-2023): VIII Mizhnar. naukovo-tekhn. konf. – Computer modeling and optimization of complex systems (KMOSS-2023): materials VIII International scientific and technical conference* (pp. 157-158). DVNZ UDKhTU. <https://udhtu.edu.ua/wp-content/uploads/2023/11/zbirnyk-tez-kmoss-2023.pdf>.
16. Rodriguez, J., Malgapo, A., Quick, J., & Huang, C. (2017). Distributed architecture of mobile GIS application using nosql database. *International Journal of Information and Electronics Engineering, 7*(6), 156–160. <https://doi.org/10.18178/ijee.2017.7.6.681>.
17. Krechowicz, A., Deniziak, S., & Lukawski, G. (2021). Highly scalable distributed architecture for nosql datastore supporting strong consistency. *IEEE Access, 9*, 69027–69043. <https://doi.org/10.1109/access.2021.3077680>.
18. Ali, A., Naem, S., Anam, S., & Ahmed, M. (2023). A State of Art Survey for Big Data Processing and NoSQL Database Architecture. *International Journal of Computing and Digital Systems, 14*(1), 297–309. <https://doi.org/10.12785/ijcds/140124>.
19. Shamsi, J. A., & Khojaye, M. A. (2021). *Big data systems: A 360-degree approach*. Taylor & Francis Group.
20. Rahman, R. U., & Tomar, D. S. (2017). Scalable security analytics framework using nosql database. *International Journal of Database Theory and Application, 10*(11), 27–46. <https://doi.org/10.14257/ijtda.2017.10.11.03>.

21. Sattar, A., Lorenzen, T., & Nallamaddi, K. (2013). Incorporating NoSQL into a database course. *ACM Inroads*, 4(2), 50–53. <https://doi.org/10.1145/2465085.2465100>.
22. Waldman, D. A., & Bowen, D. E. (1998). The acceptability of 360 degree appraisals: A customer-supplier relationship perspective. *Human Resource Management*, 37(2), 117–129. [https://doi.org/10.1002/\(sici\)1099-050x\(199822\)37:2%3C117::aid-hrm3%3E3.0.co;2-z](https://doi.org/10.1002/(sici)1099-050x(199822)37:2%3C117::aid-hrm3%3E3.0.co;2-z).
23. Abdullah A., & Arjunan T. (2023). Leveraging Advanced Machine Learning Techniques for Enhanced Intrusion and Fraud Detection in NoSQL Database Systems. *International Journal of Applied Machine Learning and Computational Intelligence*, 13(11).
24. Sadalage, P., & Fowler, M. (2013). *NoSQL distilled: A brief guide to the emerging world of polyglot persistence*. Addison-Wesley.
25. Liashenko, O.A., Lytvynov, C.N, & Solodkaia, N.A. (2019). Analiz proizvoditelnosti baz danih PostgreSQL/POSTGIS i MONGODB dlya geoprostranstvennikh zaprosov [Analysis of database performance PostgreSQL/POSTGIS i MONGODB for geo-expired queries]. *Visnyk Kremenchutskoho natsionalnoho universytetu imeni Mykhaila Ostrohradskoho –Herald of Mykhailo Ostrogradsky National University of Kremenchug*, 6(119), 60–67. <https://doi.org/10.30929/1995-0519.2019.6.60-67>.

Отримано 22.08.2024

UDC 004.65

**Oleh Reznichenko<sup>1</sup>, Oksana Liashenko<sup>2</sup>, Vyktoriia Arkhypova<sup>3</sup>**<sup>1</sup>Postgraduate Student of the Department of Information Systems

Ukrainian State University of Science and Technologies (Dnipro, Ukraine)

**E-mail:** [rov250977@gmail.com](mailto:rov250977@gmail.com). **ORCID:** <https://orcid.org/0009-0001-0194-5079><sup>2</sup>PhD in Technical Sciences,

Associate Professor of the Department of Information Systems

Ukrainian State University of Science and Technologies (Dnipro, Ukraine)

**E-mail:** [liashenko\\_o\\_a@ukr.net](mailto:liashenko_o_a@ukr.net). **ORCID:** <https://orcid.org/0000-0002-9983-5504><sup>3</sup>PhD in Technical Sciences,

Senior Research Officer at the Scientific Centre for Innovative Materials and Technologies

Associate Professor of the Department of Civil Engineering, Construction Technologies and Environmental Protection

Dnipro State Agrarian and Economic University (Dnipro, Ukraine)

**E-mail:** [arh.v.1006@gmail.com](mailto:arh.v.1006@gmail.com). **ORCID:** <https://orcid.org/0000-0002-9722-5622>**CHOICE BETWEEN RELATIONAL AND NON-RELATIONAL DATABASES**

*Creation of a large amount of data in industry, science and society has led to the problem of how to accommodate, store and process it. A separate issue is that this information is very variable, difficult to classify, and the speed of computers is becoming insufficient compared to its quantity.*

*Therefore, effective organisation of data management and the choice of storage will affect the final result of their usage. The purpose of this article is to evaluate modern database models and their use for various sectors of society, to choose a more efficient and appropriate one depending on the advantages and disadvantages. Different types of databases are used to store data, the most common of which are relational SQL and non-relational NoSQL. SQL databases store data in the form of a set of rows and columns, i.e. in a tabular form.*

*This has led to the emergence of non-relational NoSQL databases: key-value, document-oriented, columnar, and graph.*

*The analysis showed that each type of database has its advantages and disadvantages. For example, SQL is a simple, mathematical logic-based, standardised and productive form. However, the disadvantages of databases are limitations in working with unstructured data, the amount of which is growing, and scalability. NoSQLs do not store information in tabular form, have dynamic flexible schemas, and are capable of storing semi-structured and unstructured data. Compared to relational databases, they do not support complex queries, but this is compensated by their ability to store data of various types, support for horizontal scaling, and significant data processing speed. They are also capable of using aggregates and clusters. This makes it possible to use them widely for big data. Thus, we can conclude that the use of NoSQL databases is promising.*

**Keywords:** information; relational databases; non-relational databases; NoSQL; SQL.

*Fig.: 2. References: 25.*