

# Система керування базами даних та її призначення

Система **У**правління/**К**ерування **Б**азами **Д**аних (**СУБД**, **СКБД** англ. Database Management System, **DBMS**) — це набір взаємопов'язаних даних (база даних) та програм для доступу до цих даних. Вона надає можливість створення, збереження, оновлення та пошуку інформації в базах даних із контролем доступу до даних.

Система управління базами даних - це проміжне програмне забезпечення, яке дозволяє програмістам, адміністраторам баз даних (БД), програмним застосункам і кінцевим користувачам зберігати, організовувати, отримувати доступ, виконувати запити та маніпулювати даними в базі даних.

СУБД є важливими, оскільки вони надають ефективні та надійні механізми для організації, керування і використання великих обсягів даних, забезпечуючи при цьому цілісність даних та надаючи інші переваги управління даними.

В корпоративному середовищі системи управління базами даних надають адміністраторам баз даних (АБД) структуровану платформу, яка сприяє обміну даними між різними відділами, командами і програмами. СУБД надають співробітникам контрольований і організований доступ до даних, які вони можуть використовувати для сприяння інноваціям та допомагати своїй компанії підтримувати конкурентну перевагу.

## Історія

Історія систем управління базами даних налічує понад півстоліття і розпочалася в ранні 1960-ті, коли комп'ютери стали використовуватися для обробки даних. Тоді це поняття зазвичай називалося системою управління інформацією.

Першою комерційно доступною системою управління базами даних була Integrated Data Store (IDS). Вона була розроблена Чарльзом В. Бахманом і його командою в компанії General Electric (GE) наприкінці 1960-х років.

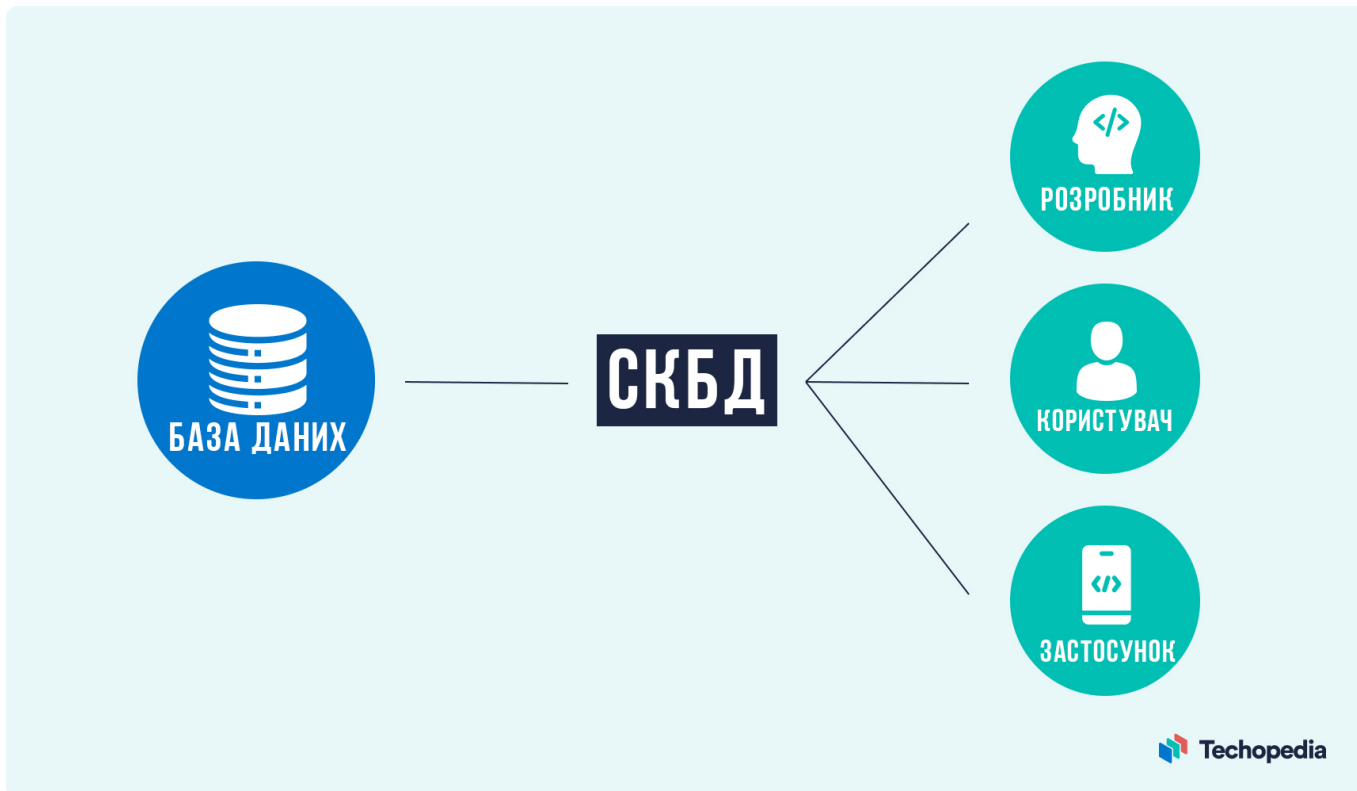
Рік	Подія
1964	Розробка першої бази даних, Integrated Data Store (IDS), Чарльзом Бахманом у компанії General Electric.
1966	IBM представляє систему управління інформацією (IMS), спільний проект з Rockwell та Caterpillar.
1970	Едгар Ф.Кодд <sup>1</sup> представляє реляційну модель в статті "Реляційна модель даних для великих спільних банків даних".
1974	Створено мову структурованих запитів (SQL).
1976	Пітер Чен <sup>2</sup> представляє модель сутність-зв'язок (ER) в своїй статті "Модель сутність-зв'язок - до єдиної точки зору на дані".
1979	Компанія Oracle випускає першу комерційну реляційну базу даних, яка використовує SQL.
1980	IBM вводить System R, реляційну систему управління базами даних на основі SQL.

Рік	Подія
1981	Представлена SQL/DS, перша повнофункціональна СУБД, яка працює на персональних комп'ютерах, компанією IBM.
1983	Випускається перша версія DB2 від IBM для мейнфреймів.
1986	Опубліковано Маніфест об'єктно-орієнтованих баз даних, що значно підтримав розвиток об'єктно-орієнтованих баз даних.
1996	Запущено PostgreSQL, одну з перших відкритих реляційних систем управління базами даних.
1998	Випуск MySQL, іншої важливої відкритої реляційної СУБД, для Windows 95 і NT.
1998	Microsoft запускає SQL Server 7.0, повністю переписавши свою СУБД.
2004	Поняття "NoSQL" набуває популярності, що призводить до нового покоління нереляційних розподілених баз даних.
2012	Amazon представляє DynamoDB, пропрієтарну нереляційну базу даних NoSQL.
2013	Випуск FoundationDB, розподіленої бази даних, призначеної для обробки великих обсягів структурованих даних.
2017	Google анонсує Spanner, глобально розподілену базу даних.
2020-ті	Продовження розвитку та інновацій в технологіях систем управління базами даних, з акцентом на бази даних, спрямовані на хмарні середовища, бази даних на краю та поліпшення інтеграції із штучним інтелектом для управління базами даних. Також бази даних блокчейн стають значущою темою інтересу.

## База даних і Система керування базою даних

Терміни "база даних" та "система управління базами даних" часто використовуються взаємозамінно в неофіційних розмовах. Це можливо через те, що коли кінцеві користувачі взаємодіють з **базою даних**, вони не мають уявлення про внутрішню **СКБД** та її основну роль у керуванні даними. Ще більше плутанини вноситься в деяких випадках, коли СКБД вбудована безпосередньо в код програми. Це робить видимість окремої системи ще менш очевидною.

Для розрізнення цих двох термінів і використання їх правильно, корисно розуміти їхні відповідні ролі та функціональність: **база даних** - це структурована колекція даних. **Система управління базами даних** - це програмне забезпечення, яке розробники, кінцеві користувачі та програми використовують для взаємодії з базою даних.



## Архітектура СКБД

Існує трирівнева система організації СКБД ANSI-SPARC, при якій існує незалежний рівень для ізоляції програми від особливостей представлення даних на нижчому рівні.

Рівні:

- Зовнішній — представлення БД з точки зору користувача.
- Концептуальний — узагальнене представлення БД, описує які дані зберігаються в БД і зв'язки між ними. Підтримує зовнішні представлення, підтримується внутрішнім рівнем.
- Внутрішній — фізичне представлення БД в комп'ютері.

*Логічна незалежність* — повна захищеність зовнішніх моделей від змін, що вносяться в концептуальну модель.

*Фізична незалежність* — захищеність концептуальної моделі від змін, які вносяться у внутрішню модель.

## Компоненти СУБД

Основним компонентом кожної СУБД є базовий двигун бази даних. Це програмне забезпечення, яке взаємодіє безпосередньо з базовою системою зберігання або файловою системою і окреслює взаємодію між модульними підкомпонентами, які дозволяють двигуну керувати та маніпулювати даними в системі баз даних. До них включаються модулі для:

**Резервного копіювання та відновлення:** ці модулі керують операціями резервного копіювання та відновлення даних, щоб захистити від втрати даних або збоїв системи. Вони

включають механізми для створення резервних копій баз даних, відновлення даних та виконання операцій відновлення.

**Контроль конкурентності та управління транзакціями:** ці модулі керують одночасним доступом до бази даних декількома користувачами або програмами. Вони обробляють механізми блокування та забезпечують консистентність даних.

**Мова доступу до бази даних:** цей тип модуля дозволяє двигуну бази даних обробляти та інтерпретувати запити або команди користувачів, написані мовою доступу за замовчуванням. Він аналізує синтаксис мови запитів, перевіряє запит щодо каталогової інформації та генерує оптимізований план виконання для отримання або маніпулювання даними за запитом.

**Мова визначення даних (Data Definition Language, DDL):** модуль DDL дозволяє користувачам визначати структуру та організацію даних. Він включає команди для створення, зміни та видалення об'єктів бази даних, таких як таблиці, види, індекси та обмеження.

У запитах SQL зазвичай використовуються DDL: CREATE, ALTER, DROP і TRUNCATE.

**Словник даних:** словник даних (також називається репозитарієм метаданих) зберігає метадані про базу даних, включаючи інформацію про структуру, відносини та властивості даних. Цей модуль використовується двигуном СУБД для забезпечення консистентності даних та виконання обмежень на унікальність.

**Мова маніпулювання даними (Data Manipulation Language, DML):** модуль DML надає команди для маніпулювання та вибору даних у базі даних. Користувачі можуть використовувати оператори DML для вставки, оновлення, видалення та запиту даних.

DML схожий на просту англійську мову і поліпшує ефективну взаємодію користувача з системою. Функціональні можливості DML організовані в командах маніпулювання, такі як:

**SELECT:** Ця команда використовується для вибору рядків з таблиці.

```
SELECT [ім'я(а) стовпця(ів)] FROM [ім'я таблиці] WHERE [умови].
```

**UPDATE:** Ця команда змінює дані одного чи декількох записів.

```
UPDATE [ім'я таблиці] SET [ім'я стовпця = значення] WHERE [умова].
```

**INSERT:** Ця команда додає один чи декілька записів до таблиці бази даних.

```
INSERT INTO [ім'я таблиці] [стовпець(и)] VALUES [значення(я)].
```

**DELETE:** Ця команда видаляє один чи декілька записів з таблиці відповідно до вказаних умов.

```
DELETE FROM [ім'я таблиці] WHERE [умова].
```

**Обробка даних та бізнес-аналітика:** ці модулі полегшують видобуток, перетворення та завантаження даних з різних джерел в окремий склад даних. Вони також підтримують онлайн-аналітичну обробку (OLAP) та засоби звітності для бізнес-аналітики.

**Індексація:** Більшість СУБД включають модулі індексації для прискорення виконання запитів за рахунок зменшення обсягу даних, які потрібно просканувати.

**Блокування:** компонент менеджера блокування СУБД відповідає за управління контролем конкурентності. Він запобігає конфліктам і забезпечує консистентність даних, переконуючись, що декілька користувачів або транзакцій не можуть одночасно змінювати одні й ті самі дані.

**Логування та аудит:** СУБД часто включають модулі для реєстрації подій бази даних та аудиту. Журнальні записи документують зміни в базі даних, включаючи вставки, оновлення та видалення, а також події системи, такі як резервне копіювання та відновлення. Аудит включає в себе моніторинг та огляд цих журналів для відстеження дій користувачів, збереження цілісності даних та допомогу в забезпеченні дотримання політики з безпеки.

**Обробка запитів:** процесор запитів приймає та інтерпретує запити користувачів, перетворює їх в оптимізований план виконання та взаємодіє з базовим двигуном бази даних для ефективного виконання запитів. Він включає підмодулі для оптимізації запитів, які враховують фактори, такі як наявність індексів, операції з'єднання та методи доступу до даних.

**Реплікація:** деякі СУБД підтримують реплікацію даних, що включає створення та підтримку кількох копій бази даних в різних місцях або на різних серверах. Реплікація поліпшує доступність даних, стійкість до збоїв та продуктивність. Вона забезпечує, що якщо одну копію бази даних неможливо отримати, дані все одно можна отримати з іншої копії.

**Безпека та авторизація:** модулі безпеки та авторизації керують доступом користувачів до бази даних та забезпечують конфіденційність та цілісність даних. Вони обробляють аутентифікацію, управління користувачами та впроваджують механізми контролю доступу на підставі **принципу найменшого привілею (Principle of Least Privilege, PoLP)** на основі ролей користувачів та вимог до роботи.

**Зберігання:** двигун бази даних в СУБД взаємодіє з двигуном зберігання для управління фізичним зберіганням даних. Двигун зберігання відповідає за обробку низькорівневих деталей того, як дані зберігаються та отримуються, тоді як двигун бази даних координує та оркеструє ці дії для оптимізації загальної роботи системи управління базою даних.

**Інтерфейси користувача:** ці модулі надають інтерфейси користувачам, які дозволяють програмістам, адміністраторам баз даних та кінцевим користувачам взаємодіяти з базою даних. Це може включати **інтерфейси командного рядка (CLI)**, **графічні інтерфейси користувача (GUI)** або **\*\*програмні інтерфейси програмування\*\* (API)** для інтеграції програмного забезпечення.

**Політики користувачів:** політики користувачів визначають та виконують контроль доступу та внутрішні політики безпеки. Вони вказують дозволи, ролі та привілеї та визначають, як користувачі можуть взаємодіяти з базою даних.

# Реляційна система керування базами даних (RDBMS) і системи керування базами даних (DBMS)

Реляційна система керування базами даних (RDBMS) - це тип системи керування базами даних (DBMS). Усі RDBMS є DBMS, але не всі системи керування базами даних є реляційними системами керування базами даних.

СКБД	Реляційні СКБД
Різні типи можуть бути використані для управління різними моделями баз даних.	Може управляти лише моделлю реляційної бази даних.
Різні типи можуть зберігати дані у різних структурах.	Завжди зберігає дані у таблицях з рядками та стовпцями.
Різні типи можуть автоматично або не автоматично забезпечувати цілісність даних.	Автоматично забезпечує цілісність даних через обмеження, такі як первинний ключ, унікальний ключ і зовнішній ключ.
Різні типи можуть використовувати різні мови або техніки для маніпулювання даними.	Завжди використовує SQL (Structured Query Language).

## Види систем управління базами даних

На початку системи управління базами даних класифікувалися як реляційні або нереляційні, залежно від їхньої структури і призначення. Якщо СУБД зберігала дані в таблицях, її називали реляційною СУБД (RDBMS). Якщо вона не зберігала дані в таблицях, то це була нереляційна СУБД, відома також як NoSQL.

Сьогодні системи управління базами даних все ще класифікуються як реляційні або нереляційні СУБД, але їх також класифікують за унікальними перевагами, які вони надають.

## Види СУБД

### Системи управління базами даних в хмарі

СУБД в хмарі, такі як Amazon Aurora, призначені для управління розподіленими даними, збереженими у віддалених центрах обробки даних хмарного постачальника.

### Системи управління базами даних з колоночною структурою

СУБД з колоночною структурою, такі як Apache Cassandra, прискорюють виконання запитів, зберігаючи дані у вигляді колонок, а не рядків. Ця схема спрощує роботу з великими наборами даних у сфері аналізу даних і бізнес-інтелекту.

### Розподілені системи управління базами даних

Функціональність розподілених систем управління базами даних, подібної до тієї, яку можна знайти в екосистемі Apache Hadoop, призначена для забезпечення цілісності даних для баз даних, які мають логічні відносини, розташовані в різних місцях або обчислювальних середовищах.

### **Системи управління базами даних графової моделі**

Системи призначені для підтримки графових баз даних, які зберігають відносини на рівні окремих записів. Системи управління базами даних графової моделі, такі як Neo4j, ідеально підходять для управління даними зі зв'язками, такими як дані соціальних медіа.

### **Системи управління базами даних ієрархічної моделі**

Призначені для підтримки баз даних, організованих за відносинами батько-дитина. Цей тип СУБД має свої коріння в обчислювальних системах головного керування і сьогоднішнє їхнє використання обмежене.

### **Системи управління базами даних гібридного типу транзакцій/аналітичного оброблення**

Системи, які спрощуються як HTAP (Hybrid transaction/analytical processing), призначені для підтримки змішаних робочих навантажень для транзакційних і аналітичних даних. Традиційні системи управління базами даних часто мають окремі системи для оброблення транзакцій в режимі реального часу (OLTP) та обробки аналітичних даних (OLAP). Системи управління базами даних типу HTAP, такі як SAP HANA і CockroachDB, надають єдину платформу, яка може обробляти обидві види завдань одночасно.

### **Системи управління базами даних в оперативній пам'яті**

Призначені для зменшення затримок шляхом використання основної пам'яті для управління та зберігання даних. IMDBMS, такі як Volt Active Data і інші, значно прискорюють отримання даних і покращують загальну продуктивність системи.

### **Системи управління базами даних об'єктно-орієнтованої моделі (OODBMS)**

СУБД об'єктно-орієнтованої моделі призначені для управління складними структурами даних у вигляді об'єктів зберігання.

### **Системи управління базами даних типу NewSQL**

СУБД типу NewSQL, такі як PostgreSQL, надають можливості масштабування і продуктивності баз даних NoSQL, при цьому зберігаючи ACID-властивості традиційних реляційних баз даних. Цей тип СУБД призначений для розподілених середовищ великого масштабу і може обробляти





високовитратні транзакційні завдання.

## Системи управління базами даних часових рядів

СУБД часових рядів, такі як InfluxDB, оптимізують зберігання, відновлення і аналіз даних з відмітками часу. Їх часто використовують для підтримки фінансового аналізу і систем моніторингу Інтернету речей (IoT).

## Відомі системи керування базами даних

	MS Access- легка реляційна система управління базами даних (RDBMS), включена в пакет Microsoft Office і Office 365.
	Amazon RDS - хмарна СУБД, яка пропонує двигуни для управління базами даних MySQL, Oracle, SQL Server, PostgreSQL та Amazon Aurora.
	Apache Cassandra - відкрита розподілена система управління базами даних, відома своєю здатністю обробляти величезні обсяги даних.
	Filemaker - реляційна СКБД баз даних, раніше Claris (Кларис), дочірньої компанії Apple Inc. Додаток поєднує ядро бази даних з графічним інтерфейсом користувача (GUI) і функціями безпеки, що дозволяє користувачам змінювати базу даних за допомогою перетягування нових елементів в макети, екрани або форми
	Google Cloud Spanner - глобально розподілена, горизонтально масштабована та сильно консистентна служба реляційних баз даних, пропонована Google Cloud.
	IBM Db2 - родина реляційних систем управління базами даних, розроблених IBM, які пропонуються в різних редакціях для різних середовищ і завдань.
	MariaDB - відкрита реляційна база даних, відгалуження MySQL.
	Microsoft Azure SQL Database - хмарна реляційна база даних, надана Microsoft Azure, яка пропонує повністю керовані SQL-бази даних.
	MongoDB - популярна система управління базами даних NoSQL, яка використовує схему, орієнтовану на документи, для забезпечення високої масштабованості та гнучкості.
	MySQL - відкрита реляційна система управління базами даних (RDBMS), належить Oracle.
	Oracle - пропрієтарна <sup>3)</sup> RDBMS, оптимізована для гібридних хмарних архітектур.

	PostgreSQL - відкрита реляційна система управління базами даних, відома своєю надійністю, масштабованістю та великим набором функцій.
	SAP HANA - колоночна RDBMS, оптимізована для операцій з реальним часом і високопродуктивних аналітичних завдань.
	SQL Server - корпоративна реляційна система управління базами даних від Microsoft, здатна обробляти дуже великі обсяги даних і запитів до бази даних.
	SQLite - легкий файловий реляційний двигун баз даних, який широко використовується в вбудованих системах і мобільних додатках.
	Teradata - потужний SQL-двигун, який надає масштабовані рішення для управління та аналізу великих обсягів даних.

## Переваги використання систем управління базами даних (СУБД)

Системи управління базами даних (СУБД) особливо важливі в ситуаціях, де кілька користувачів або додатків взаємодіють з однією і тією ж базою даних одночасно. СУБД забезпечують захист від конфліктів і помилок завдяки механізмам управління паралельністю, які гарантують, що навіть в умовах великого навантаження цілісність даних залишається недоторканою.

Ще одною перевагою є те, що системи управління базами даних пропонують широкий спектр засобів, механізмів і функцій забезпечення безпеки. Адміністратори можуть визначати правила контролю доступу, призначати ролі користувачів і вказувати дозволи, щоб гарантувати, що лише авторизовані особи можуть вводити, отримувати доступ і маніпулювати даними.

Оскільки СУБД надають засоби ведення журналу та реєстрації для відстеження та моніторингу доступу до даних та їх модифікацій, вони є корисними засобами для виконання вимог щодо відповідності.

Наприклад, СУБД може допомогти адміністраторам керувати життєвим циклом даних, виконуючи політики збереження, архівації та остаточного видалення даних. СУБД також може допомогти здійснювати контроль за конфіденційністю, надаючи механізми, які анонімізують або шифрують конфіденційні дані.

## Джерела

- [Система управління базами даних](#)
- [What is a DBMS? Uses, Types, Components, Examples & History](#)
- [What is Data Manipulation Language \(DML\)? - Definition from Techopedia](#)

1)

Едгар Франк «Тед» Кодд — британський інформатик, який, працюючи у IBM, винайшов, окрім іншого, реляційну модель даних, теоретичну основу для реляційних баз даних.

2)

Пітер Пін-Шан Чен — тайванський американський вчений-інформатик. Він є видатним науковцем і викладачем Університету Карнегі-Меллона та почесним професором-емеритом

кафедри LSU. Він відомий розробкою моделі сутність-зв'язок у 1976 році.

3)

Власницьке програмне забезпечення, або Пропрієтарне програмне забезпечення, (від англ. proprietary software) — це програмне забезпечення, на яке зберігаються як немайнові, так і майнові авторські права. Отримавши або придбавши таке програмне забезпечення, користувач отримує обмежені права користування ним: може бути заборонено або закрито доступ до коду (вивчення), внесення змін, тиражування, розповсюдження та перепродаж. Програмне забезпечення вважається власницьким або закритим, якщо наявне хоча б одне з перелічених обмежень.

From:

<https://library.vpuhluhiv.com.ua/> - **Wiki Глухівського ВПУ**

Permanent link:

[https://library.vpuhluhiv.com.ua/subjects:basic:informatika:base:systema\\_keruvannia\\_bazamy\\_danyh\\_ih\\_pryznahennia](https://library.vpuhluhiv.com.ua/subjects:basic:informatika:base:systema_keruvannia_bazamy_danyh_ih_pryznahennia)

Last update: **10.09.2023 20:11**

