1. Основные понятия БД

База данных (БД) – совокупность связанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования, независимая от прикладных программ

Информационная система (ИС) –  система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы, которые обеспечивают и распространяют информацию

Вычислительная ситсема (ВС) – это взаимосвязанная совокупность аппаратных средств вычислительной техники и программного обеспечения, предназначенная для обработки информации

Банк данных (БнД) -  является современной формой организации хранения и доступа к информации.

Система управления базой данных (СУБД) – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных

Словарь данных – набор доступных для выборки всем пользователям базы данных системных таблиц, в которых хранятся метаданные (данныео данных).

Администратор БД - специалист, отвечающий за обслуживание систем управления баз данных. Главная его обязанность — обеспечение постоянной работы сервера и доступа пользователей к нужной информации.

1. Функции СУБД

**Администрирование базы данных**. В процессе эксплуатации БД администратор обычно следит за ее функционированием, обеспечивает защиту от несанкционированного доступа к хранимым данным, вносит изменения в структуру базы, контролирует достоверность информации в ней.

**Непосредственное управление данными во внешней памяти.** Эта функция предоставляет пользователю возможность выполнения основных операций с данными – хранение, извлечение и обновление информации.

**Управление буферами оперативной памяти.** СУБД обычно работают с БД значительного размера; по крайней мере этот размер обычно существенно больше доступного объема оперативной памяти. При обращении к любому элементу данных будет производиться обмен с внешней памятью, то вся система будет работать со скоростью устройства внешней памяти. Решением является буферизация данных в оперативной памяти. Однако этого недостаточно для целей СУБД. Поэтому в развитых СУБД поддерживается собственный набор буферов оперативной памяти.

**Управление транзакциями.** Транзакция – это последовательность операций над БД, которые рассматриваются СУБД как единое целое и позволяют добавлять, удалять или обновлять сведения о некотором объекте в базе

1. Классификации СУБД

**По**[**модели данных**](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)**:**

* [Иерархические](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Сетевые](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Реляционные](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Объектно-ориентированные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Объектно-реляционные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

**По степени распределённости:**

* Локальные СУБД (все части локальной СУБД размещаются на одном компьютере)
* Распределённые СУБД (части СУБД могут размещаться не только на одном, но на двух и более компьютерах).

**По способу доступа к БД:**

* [Файл-серверные](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80)

В файл-серверных СУБД файлы данных располагаются централизованно на [файл-сервере](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80). СУБД располагается на каждом клиентском компьютере (рабочей станции). Доступ СУБД к данным осуществляется через [локальную сеть](https://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%BA%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C). Синхронизация чтений и обновлений осуществляется посредством файловых блокировок.

Преимуществом этой архитектуры является низкая нагрузка на процессор файлового сервера.

Недостатки: потенциально высокая загрузка локальной сети; затруднённость или невозможность [централизованного](https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BD%D1%82%D1%80%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [управления](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F); затруднённость или невозможность обеспечения таких важных характеристик, как высокая [надёжность](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%B4%D1%91%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D1%8C), [высокая доступность](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%B0%D1%8F_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BD%D0%BE%D1%81%D1%82%D1%8C) и высокая [безопасность](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B5%D0%B7%D0%BE%D0%BF%D0%B0%D1%81%D0%BD%D0%BE%D1%81%D1%82%D1%8C). Применяются чаще всего в локальных приложениях, которые используют функции управления БД; в системах с низкой интенсивностью обработки данных и низкими пиковыми нагрузками на БД.

На данный момент файл-серверная технология считается устаревшей, а её использование в крупных информационных системах — недостатком[[2]](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85#cite_note-2).

Примеры: [Microsoft Access](https://ru.wikipedia.org/wiki/Microsoft_Access), [Paradox](https://ru.wikipedia.org/wiki/Paradox), [dBase](https://ru.wikipedia.org/wiki/DBase), [FoxPro](https://ru.wikipedia.org/wiki/FoxPro), [Visual FoxPro](https://ru.wikipedia.org/wiki/Visual_FoxPro).

* [Клиент-серверные](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

Клиент-серверная СУБД располагается на сервере вместе с БД и осуществляет доступ к БД непосредственно, в монопольном режиме. Все клиентские запросы на обработку данных обрабатываются клиент-серверной СУБД централизованно.

Недостаток клиент-серверных СУБД состоит в повышенных требованиях к серверу.

Достоинства: потенциально более низкая загрузка локальной сети; удобство централизованного управления; удобство обеспечения таких важных характеристик, как высокая надёжность, высокая доступность и высокая безопасность.

Примеры: [Oracle Database](https://ru.wikipedia.org/wiki/Oracle_Database), [Firebird](https://ru.wikipedia.org/wiki/Firebird), [Interbase](https://ru.wikipedia.org/wiki/Interbase" \o "Interbase), [IBM DB2](https://ru.wikipedia.org/wiki/IBM_DB2), [Informix](https://ru.wikipedia.org/wiki/Informix), [MS SQL Server](https://ru.wikipedia.org/wiki/MS_SQL_Server), [Sybase Adaptive Server Enterprise](https://ru.wikipedia.org/wiki/Sybase#%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%86%D0%B8%D1%8F), [PostgreSQL](https://ru.wikipedia.org/wiki/PostgreSQL), [MySQL](https://ru.wikipedia.org/wiki/MySQL), [Caché](https://ru.wikipedia.org/wiki/Cach%C3%A9" \o "Caché), [ЛИНТЕР](https://ru.wikipedia.org/wiki/%D0%9B%D0%98%D0%9D%D0%A2%D0%95%D0%A0).

[Встраиваемые](https://ru.wikipedia.org/wiki/%D0%92%D1%81%D1%82%D1%80%D0%B0%D0%B8%D0%B2%D0%B0%D0%B5%D0%BC%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

Встраиваемая СУБД — СУБД, которая может поставляться как составная часть некоторого программного продукта, не требуя процедуры самостоятельной [установки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%81%D1%82%D0%B0%D0%BB%D0%BB%D1%8F%D1%86%D0%B8%D1%8F_(%D0%9F%D0%9E)). Встраиваемая СУБД предназначена для локального хранения данных своего приложения и не рассчитана на коллективное использование в сети.

Физически встраиваемая СУБД чаще всего реализована в виде [подключаемой библиотеки](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)). Доступ к данным со стороны приложения может происходить через [SQL](https://ru.wikipedia.org/wiki/SQL) либо через специальные [программные интерфейсы](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81).

Примеры: [OpenEdge](https://ru.wikipedia.org/w/index.php?title=OpenEdge&action=edit&redlink=1" \o "OpenEdge (страница отсутствует)), [SQLite](https://ru.wikipedia.org/wiki/SQLite), [BerkeleyDB](https://ru.wikipedia.org/wiki/BerkeleyDB" \o "BerkeleyDB), [Firebird](https://ru.wikipedia.org/wiki/Firebird) Embedded, [Microsoft SQL Server Compact](https://ru.wikipedia.org/wiki/Microsoft_SQL_Server_Compact), [ЛИНТЕР](https://ru.wikipedia.org/wiki/%D0%9B%D0%98%D0%9D%D0%A2%D0%95%D0%A0).

1. Связи

Один-ко-многим, много-ко-многим, один-к-одному

1. Первичный ключ таблицы. Свойства Primary key (РК).

Первичный ключ - поле или набор полей, однозначно (уникально) идентифицирующих запись.

* *Уникальность.* В таблице нет двух разных строк с одинаковыми значениями в нашем потенциальном ключе.
* *Неизбыточность.* Нельзя убрать один из столбцов из ключа, так, чтобы он не потерял уникальности.

1. Нормализация базы данных. Индексирование

**Нормализация** — это процесс организации данных в базе данных, включающий создание таблиц и установление отношений между ними в соответствии с правилами, которые обеспечивают защиту данных и делают базу данных более гибкой, устраняя избыточность и несогласованные зависимости.

**Первичный индекс** - это такой специальный массив-указатель порядка записей, когда файл данных последовательно упорядочивается по полю ключа упорядочения, а на основе поля ключа упорядочения создается поле индексации, которое гарантированно имеет уникальное значение в каждой записи.

**Индекс кластеризации** - это такой специальный массив-указатель порядка записей, когда файл данных последовательно упорядочивается по неключевому полю, и на основе этого неключевого поля формируется поле индексации, поэтому в файле может быть несколько записей, соответствующих значению этого поля индексации. Неключевое поле называется атрибутом кластеризации.

**Вторичный индекс** - это индекс, который определен на поле файла данных, отличном от поля, по которому выполняется упорядочение.

1. Целостность базы данных. Каскадное обновление и удаление записей.

Обеспечение целостности данных гарантирует качество данных в таблице. Например, если служащему присвоен идентификатор **123**, база данных не должна позволять другим служащим иметь такое же значение идентификатора. Если существует столбец **employee\_rating**, в котором значения должны находиться в диапазоне от **1** до **5**, база данных не должна сохранять в нем значения, лежащие вне этого диапазона.

Если при определении связи установить флажок Каскадное удаление связанных записей, любое удаление записи в главной таблице приведет к автоматическому удалению связанных записей в подчиненной таблице.

1. Первая нормальная форма

Отношение находится в 1НФ, если все его атрибуты являются простыми, все используемые домены должны содержать только скалярные значения. Не должно быть повторений строк в таблице.

1. Вторая нормальная форма

Отношение находится во 2НФ, если оно находится в 1НФ и каждый не ключевой атрибут неприводимо зависит от Первичного Ключа(ПК).  
  
Неприводимость означает, что в составе потенциального ключа отсутствует меньшее подмножество атрибутов, от которого можно также вывести данную функциональную зависимость.

1. Третья нормальная форма

Отношение находится в 3НФ, когда находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

1. Основные типы данных в sql и их свойства

|  |  |  |
| --- | --- | --- |
| BINARY | 1 байт на символ | В таком поле могут храниться данные любого типа. Преобразование данных (например, в текст) не производится. От того, как вводятся данные в поле BINARY, зависит отображение выходных данных. |
| BIT | 1 байт | Значения Yes и No, а также поля, содержащие только одно из двух значений. |
| TINYINT | 1 байт | Целое число от 0 до 255. |
| MONEY | 8 байт | Масштабируемое целое число в диапазоне от -922 337 203 685 477,5808 до 922 337 203 685 477,5807. |
| DATETIME (см. DOUBLE) | 8 байт | Значение даты и времени, относящееся к годам с 100 по 9999. |
| UNIQUEIDENTIFIER | 128 бит | Уникальный идентификационный номер, используемый для удаленного вызова процедур. |
| REAL | 4 байта | Значение одиночной точности с плавающей запятой в диапазоне от -3,402823E38 до -1,401298E-45 для отрицательных значений и от 1,401298E-45 до 3,402823E38 для положительных значений и 0. |
| FLOAT | 8 байт | Значение двойной точности с плавающей запятой в диапазоне от -1,79769313486232E308 до -4,94065645841247E-324 для отрицательных значений и от 4,94065645841247E-324 до 1,79769313486232E308 для положительных значений и 0. |
| SMALLINT | 2 байта | Короткое целое число в диапазоне от -32 768 до 32 767. |
| INTEGER | 4 байта | Длинное целое число в диапазоне от -2 147 483 648 до 2 147 483 647. |
| DECIMAL | 17 байт | Точный числовой тип данных, включающий значения от 1028 - 1 до -1028 - 1. Можно определить как точность (1 - 28), так и масштаб (точность определена как 0). Точность и масштаб по умолчанию составляют 18 и 0 соответственно. |
| TEXT | 2 байта на символ (см. примечание) | От 0 до 2,14 ГБ. |
| IMAGE | Любой | От 0 до 2,14 ГБ. Используется для объектов OLE. |
| CHARACTER | 2 байта на символ (см. примечание) | От 0 до 255 знаков. |

1. Оператор SELECT

SELECT определяет список возвращаемыех столбцов (как существующих, так и вычисляемых), их имена, ограничения на уникальность строк в возвращаемом наборе, ограничения на количество строк в возвращаемом наборе;

FROM задаёт табличное выражение, которое определяет базовый набор данных для применения операций, определяемых в других предложениях оператора;

WHERE задает ограничение на строки табличного выражения из предложения FROM;

GROUP BY объединяет ряды, имеющие одинаковое свойство с применением агрегатных функций

HAVING выбирает среди групп, определенных параметром GROUP BY

ORDER BY задает критерии сортировки строк; отсортированные строки передаются в точку вызова.

1. Оператор CREATE и DROP

DROP DATABASE [IF EXISTS] db\_name

DROP DATABASE удаляет все таблицы в базе данных и удаляет саму базу данных. Если выполнили DROP DATABASE на символически связанной базе данных, связь и первоначальная база данных будут удалены

CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl\_name [(create\_definition,...)]

[table\_options] [select\_statement]

1. Оператор INSERT

**INSERT** — оператор языка [SQL](https://ru.wikipedia.org/wiki/SQL), который позволяет добавить строки в таблицу, заполняя их значениями. Значения можно вставлять перечислением с помощью слова values и перечислив их в круглых скобках через запятую или оператором [select](https://ru.wikipedia.org/wiki/Select_(SQL)" \o "Select (SQL)).

INSERT INTO name\_table (coloumn\_name\_1) VALUES ('$znachenie\_1)

1. Оператор UPDATE

UPDATE books SET books.amount = books.amount - 1 WHERE books.id = '$id\_b'

1. Оператор UNION

В языке [SQL](https://ru.wikipedia.org/wiki/SQL) ключевое слово **UNION** применяется для объединения результатов двух SQL-запросов в единую [таблицу](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)), состоящую из схожих [строк](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D1%82%D1%80%D0%BE%D0%BA%D0%B0_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)&action=edit&redlink=1). Оба запроса должны возвращать одинаковое число [столбцов](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D1%82%D0%BE%D0%BB%D0%B1%D0%B5%D1%86_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)&action=edit&redlink=1) и совместимые [типы данных](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в соответствующих столбцах

Это сведение таблиц