1. Дайте определение понятию «база данных» (БД).

База данных – совокупность взаимосвязанных данных.

1. Дайте определение понятию «система управления базой данных» (СУБД).

СУБД - программная реализация технологии хранения, извлечения, обновления и обработки данных в базе данных.

1. Назовите основные компоненты СУБД.

1) ядро

2) процессор языка базы данных,

3) подсистему поддержки времени исполнения

4) сервисные программы

1. Какие БД называют системными?

БД, которые содержать метаданные, используемые для управления системой.

1. Какие БД называются реляционными? Назовите основные признаки реляционных баз данных.

Реляционная база данных представляет собой набор взаимосвязанных таблиц(сущностей). Таблицы(отношения) состоят из колонок(атрибутов) и строк (кортежей).

Основные признаки реляционных баз данных: тип данных, домен, атрибут, кортеж, отношение, первичный ключ.

1. Что такое ***OLTP***?

OLTP, транзакционная система — обработка транзакций в реальном времени. Способ организации БД, при котором система работает с небольшими по размерам транзакциями, но идущими большим потоком, и при этом клиенту требуется от системы минимальное время отклика.

1. Что такое ***OLAP***?

OLAP (англ. online analytical processing, интерактивная аналитическая обработка) — технология обработки данных, заключающаяся в подготовке суммарной (агрегированной) информации на основе больших массивов данных, структурированных по многомерному принципу.

1. Поясните понятие «архитектура клиент-сервер».

«Клиент — сервер» — вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами. Фактически клиент и сервер — это программное обеспечение.

1. Поясните понятие «сервис Windows».

Службы ОС Windows — приложения, автоматически запускаемые системой при запуске Windows и выполняющиеся вне зависимости от статуса пользователя.

1. Поясните понятие «программный интерфейс СУБД».

Представление БД в виде набора реляционных объектов (таблиц, представлений, ограничений целостности и пр.).

1. Поясните понятие «интерфейс командной строки».

Разновидность текстового интерфейса, в котором инструкции компьютеру даются в основном путём ввода с клавиатуры текстовых строк (команд)

1. Расшифруйте SQL и объясните, что это такое.

Язык SQL (Structured Query Language, язык структурированных запросов) – специализированный язык, предназначенный для написания запросов к реляционной БД

1. Поясните понятие «декларативный язык». Приведите примеры декларативных языков.



SQL, html

1. Поясните понятие «процедурный язык». Приведите примеры процедурных языков.

Задачи разбиваются на шаги и решаются шаг за шагом. Basic, C

1. Перечислите группы операторов языка SQL.

**DDL** - Data Definition Language - язык определения данных

**DML** - Data Manipulation Language - язык манипулирования данными

**TCL** - Transaction Control Language - язык управления транзакциями

**DCL** - Data Control Language - язык управления данными

1. Назовите основные операторы каждой группы.

**DDL:** CREATE, ALTER, DROP

**DML:** SELECT, INSERT, DELETE, UPDATE

**TCL:** BEGIN TRAN, SAVE TRAN, COMMIT TRAN, ROLLBACK TRAN

**DCL:** GRANT, REVOKE, DENY

1. Поясните понятие «ограничения целостности». Перечислите все известные вам типы ограничений целостности.

**Ограничения целостности** - это правила, применяемые к столбцам данных таблицы.

Data type, not null, default, foreign key, primary key, unique, check

1. Перечислите все секции оператора SELECT в порядке их выполнения. Поясните назначение каждой секции.

FROM источник

WHERE условие

GROUP BY группировка

HAVING условие

SELECT подзапрос

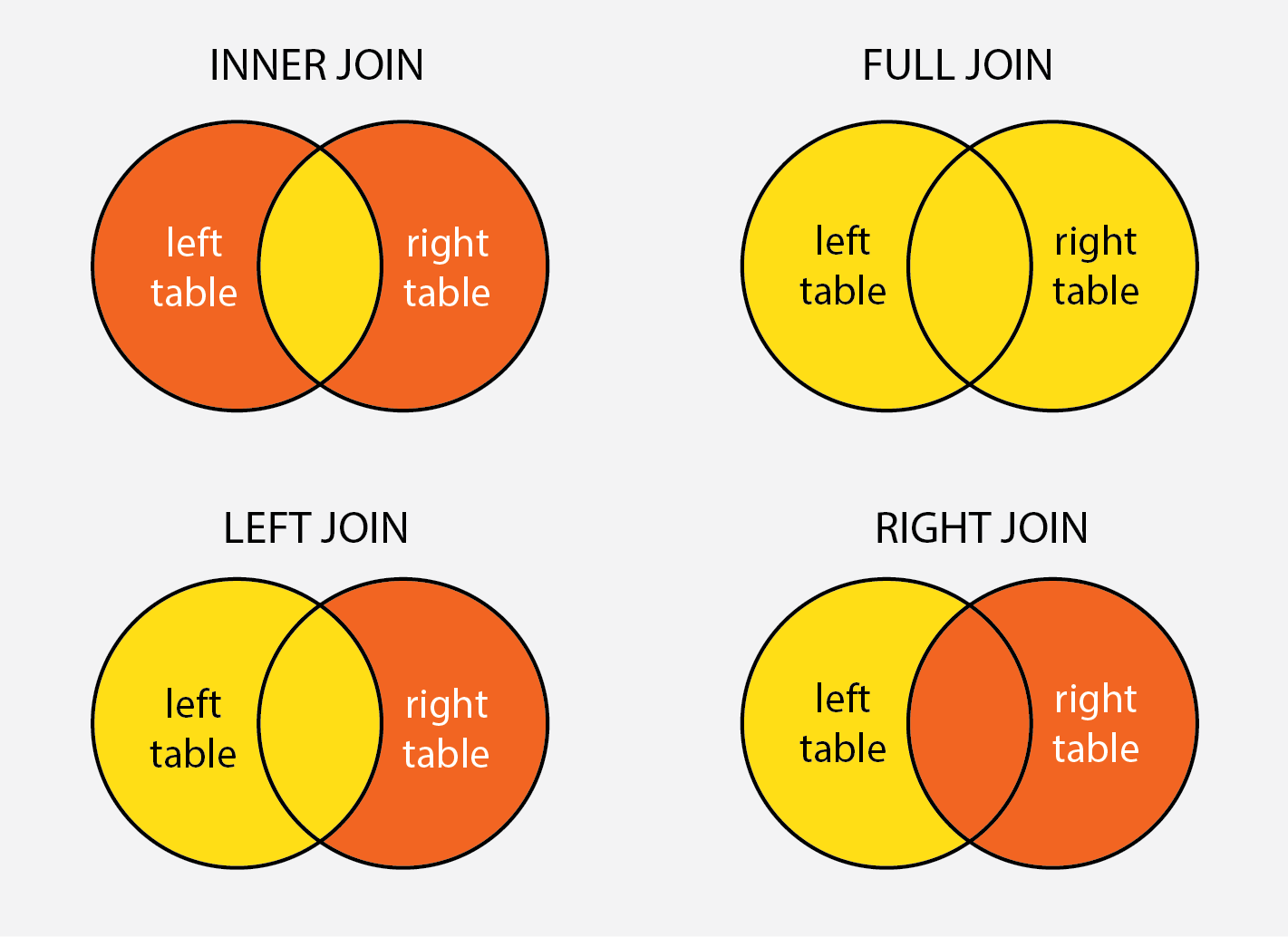
DISTINCT избавление от повторяющихся строк

ORDER BY упорядочивание

TOP отбор результата

INTO формирование и заполнение таблицы

1. Перечислите все способы соединения таблиц в SELECT-запросе? Назовите коммутативные способы соединения таблиц.



Оператор UNION, INTERSECT, INNER JOIN является коммутативным: результат не зависит от порядка объединяемых наборов.

1. Поясните понятия «подзапрос», «некоррелируемый подзапрос» и «коррелируемый подзапрос».

**Подзапрос** – это SELECT-запрос, который выполняется в рамках другого запроса.

**Коррелируемый подзапрос** зависит от внешнего запроса и выполняется для каждой строки результирующего набора.

**Независимый подзапрос** не зависит от внешнего запроса и выполняется только один раз, но результат его выполнения подставляется в каждую строку результирующего набора.

1. Поясните понятие «агрегатная функция». Перечислите известные вам агрегатные функции.

**Агрегатные функции** вычисляют некоторые скалярные значения в наборе строк.

**AVG**: вычисляет среднее значение

**SUM**: вычисляет сумму значений

**MIN**: вычисляет наименьшее значение

**MAX**: вычисляет наибольшее значение

**COUNT**: вычисляет количество строк в запросе

1. Поясните применение конструкций ***IN, ALL, ANY, BETWEEN, NОТ, IS NULL, LIKE, EXISTS*** в секции WHERE.

IN формирует логическое значение «истина» в том случае, если значение, указанное слева от ключевого слова IN, равно хотя бы одному из значений списка, указанного справа.

ALL формирует истинное значение в том случае, если значение, стоящее слева больше или равно каждому значению в списке, указанном справа.

ANY формирует истинное значение в том случае, если значение, стоящее слева, больше или равно хотя бы одному значению в списке, указанном справа.

BETWEEN Диапазон значений

NOT Меняет значение на противоположное

IS NULL Истина, если значение NULL

LIKE Сравнение строк с помощью шаблона

EXISTS Проверка наличия строк в таблице

1. Перечислите этапы обработки SQL-запроса.

1. Синтаксический разбор (parsing).

2. Компиляция запроса (query compilation).

3. Оптимизация запроса (query optimization).

4. Выполнение запроса (query execution).

1. Поясните понятия «план запроса», «стоимость запроса».

**План запроса** - алгоритм выполнения SQL-запроса.

При построении плана запроса для каждого шага вычисляется числовая величина, пропорциональная продолжительности выполнения шага, называемая **стоимостью**.

1. Поясните понятия «индекс».

**Индекс** – это объект базы данных, позволяющий **ускорить поиск** в определенной таблице, так как при этом данные организуются в виде сбалансированного бинарного дерева поиска.

1. Перечислите известные вам типы индексов.

Индексы бывают кластеризованные, некластеризованные, уникальные, неуникальные, покрытия, фильтруемые, составные.

1. В каких случаях и какого типа индексы создаются автоматически?

Обычно кластеризованные индексы создаются автоматически при создании таблицы если в ней присутствует первичный ключ (ограничение PRIMARY KEY).

1. Поясните понятие «фрагментация индекса».

Операции добавления и изменения строк базы данных могут повлечь образование неиспользуемых фрагментов в области памяти индекса. Процесс образования неиспользуемых фрагментов памяти называется фрагментацией.

1. Сколько у одной таблицы может быть кластеризованных индексов и почему?

Существует только один кластеризованный индекс для каждой таблицы, так как строки данных могут храниться в единственном порядке.

1. Поясните назначение процедур перестройки и реорганизации индексов. В чем разница?

Реорганизация (REORGANIZE) выполняется значительно быстрее, но фрагментация будет убрана только из листовых узлов индексного дерева. Рекомендуется выполнять при уровне фрагментации 10–20%.

Операция перестройки (REBUILD) затрагивает все узлы дерева, поэтому после ее выполнения степень фрагментации равна нулю. Рекомендуется выполнять при уровне фрагментации, превышающем 20%.

1. Поясните понятие «представление». С помощью какого оператора создается представление?

Представление – это объект базы данных, представляющий собой поименованный SELECT-запрос, который хранится в базе данных. Представление создается с помощью оператора CREATE, удаляется с помощью оператора DROP и изменяется с помощью ALTER.

1. В каких случаях к представлению применимы операторы INSERT, DELETE, UPDATE?

При создании представлений, позволяющих выполнять операции INSERT, DELETE и UPDATE, базовый SELECT-запрос должен удовлетворять правилам:

· запрос не должен содержать секцию группировки GROUP BY;

· запрос не должен применять агрегатные функции, опции DISTINCT и TOP, операторы UNION, INTERSECT и EXCEPT;

· в SELECT-списке запроса не должно быть вычисляемых значений;

· в секции FROM запроса должна указываться только одна таблица.

1. Какое дополнительное свойство приобретает представление с опцией WITH CHECK OPTION?

Чтобы операция вставки не могла осуществиться в том случае, когда информация не удовлетворяет условию, записанному в секции Where, то следует создавать представление с опцией WITH CHECK OPTION.

1. Чем отличается оператор TRUNCATE от оператора DELETE?

Команда **DELETE** удаляет записи из таблицы, которые удовлетворяют критерию WHERE. После **Delete** возможен откат, команда DML.

**TRUNCATE** удаляет все данные из таблицы. Откат (rollback) после **TRUNCATE** невозможен, команда DDL

1. Поясните понятие «курсор». Перечислите типы курсоров известные вам.

**Курсор** является программной конструкцией, которая дает возможность пользователю обрабатывать строки результирующего набора запись за записью. Курсоры бывают *локальные* и *глобальные* (по умолчанию), *статические* и *динамические* (по умолчанию).

1. Поясните схему работы с курсором.

DECLARE -> OPEN ->FETCH -> @@FETCH\_STATUS. -> CLOSE.

Если курсор глобальный, то он должен быть освобожден с помощью оператора DEALLOCATE.

1. Поясните назначение конструкции CURRENT OF при работе с курсором.

Позволяет удалять или изменять строки в таблице, соответствующие текущей позиции курсора в результирующем наборе.

1. Поясните понятия «транзакция», «фиксация транзакции», «откат транзакции».

**Транзакция** – механизм БД, позволяющий объединять несколько операторов, изменяющих БД таким образом, чтобы при выполнении этой совокупности операторов они или все выполнились, или все не выполнились.

**Фиксация транзакции** - это запись на диск изменений в базе данных, которые были сделаны в процессе выполнения транзакции.

Если некоторые из операций завершены, но с ошибками, а другие без, системы обработки транзакций дает команду на «**откат**» всех операций транзакции (в том числе удачных), что означает стирание всех следов операции и восстановление системы до согласованного известного состояния, которое было до начала процесса транзакции.

1. Расшифруйте и поясните аббревиатуру ACID.

**атомарность** (операторы изменения БД, включенные в транзакцию, либо выполнятся все, либо не выполнится ни один); (все или ничего)

**согласованность** (транзакция должна фиксировать новое согласованное состояние БД);

**изолированность** (отсутствие взаимного влияния параллельных транзакций на результаты их выполнения);

**долговечность** (изменения в БД, выполненные и зафиксированные транзакцией, могут быть отменены только с помощью новой транзакции).

1. Поясните понятия «режим автофиксации», «режим неявной транзакции», «режим явной транзакции».

**Режим автофиксации** означает, что каждый запрос к базе данных, который вы выполняете, неявно заключается в транзакцию, если СУБД их поддерживает.

**Неявные транзакции**: Новая транзакция неявно начинается, когда предыдущая транзакция завершена, но каждая транзакция явно завершается инструкцией COMMIT или ROLLBACK.

**Явные транзакции**: Каждая транзакция явно начинается с инструкции BEGIN TRANSACTION и явно заканчивается инструкцией COMMIT или ROLLBACK.

1. Перечислите известные вам уровни изолированности в порядке их усиления.

READ UNCOMMITED, READ COMMITED, REPEATABLE READ, SNAPSHOT, SERIALIZABLE

1. Поясните понятие «неподтвержденное чтение», «неповторяющееся чтение», «фантомное чтение».

1.Чтение данных, добавленных или изменённых транзакцией, которая впоследствии не подтвердится (откатится).

2. Ситуация, когда при повторном чтении в рамках одной транзакции ранее прочитанные данные оказываются изменёнными.

3. Ситуация, когда при повторном чтении в рамках одной транзакции одна и та же выборка дает разные множества строк.

1. Поясните понятия «хранимая процедура», «входной параметр», «выходной параметр», «значение, возвращаемое к точке вызова», «позиционная форма передачи параметров», «параметрическая форма передачи параметров», «системная хранимая процедура».

Хранимая процедура – это поименованный код T-SQL.

Хранимая процедура может принимать входные и формировать выходные параметры, а результатом ее выполнения может быть целочисленное значение, возвращаемое с помощью оператора RETURN.

Позиционная форма передачи – параметры передаются с помощью символа ?

При вызове хранимой процедуры применяется параметрическая форма передачи параметров.

Системная хранимая процедура – процедура поставляемые разработчиками СУБД.

1. Поясните понятия «скалярная функция», «встроенная табличная функция», «многооператорная хранимая функция».

**Скалярная** функция возвращает единственное значение.

Во **встроенных** функциях с табличным значением возвращаемое значение TABLE.

Многооператорная функция - возвращает таблицу, созданную одним или несколькими операторами T-SQL.

1. Назовите отличия хранимых процедур от функций.

Отличие функций от хранимых процедур в ограничениях, накладываемых на код функции, в форме представления результата работы, а также в способе вызова. В функции не допускается применение DDL-операторов, DML-операторов, изменяющих БД (INSERT, DELETE, UPDATE), конструкций TRY/CATCH, а также использование транзакций.

1. Поясните понятия «DDL-триггер», «DML-триггер».

Каждый DML-триггер связан с таблицей или представлением и предназначен для обработки одного или нескольких событий, соответствующих трем операторам: INSERT, UPDATE и DELETE.

DDL – перехват команд DDL (два уровня: уровень сервера и уровень бд).

1. Поясните понятия «триггер уровня оператора», «триггер уровня строки», «событие, активизирующее триггер».

Триггер уровня оператора вызывается не для каждой модифицируемой записи, а один раз на изменение таблицы.

Триггер уровня строки вызывается для каждой модифицируемой записи.

Событие, активизирующее триггер – событие приводящее к срабатыванию триггера.

1. Поясните смысл выражения «триггер является частью транзакции».

Ошибка в триггере откатывает операцию, изменения таблиц в триггере становятся частью транзакции.

1. Поясните понятия «XML», «XML-Schema», «W3C», «тег», «атрибут», «корневой тег», «XML-узел», «XML-элемент», «XML-документ», «элементная форма XML-документа», «атрибутная форма XML-документа»;

XML ( eXtensible Markup Language) — расширяемый [язык разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8).

Консо́рциум Всеми́рной паути́ны ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) World Wide Web Consortium, W3C) — организация, разрабатывающая и внедряющая технологические стандарты для [Всемирной паутины](https://ru.wikipedia.org/wiki/%D0%92%D1%81%D0%B5%D0%BC%D0%B8%D1%80%D0%BD%D0%B0%D1%8F_%D0%9F%D0%B0%D1%83%D1%82%D0%B8%D0%BD%D0%B0).

XML Schema — язык описания структуры XML-документа.

Тег –элемент языка разметки гипертекста.

**атрибут** — пара имя-значение, добавляемые в открывающий тег после названия элемента.

**XML-узел** – любая часть XML документа.

**XML элемент** — это все от (и включая) начального тега элемента до (и включая) конечного тега элемента.

XML-документ представляет собой обычный текстовый файл, в котором при помощи специальных маркеров создаются элементы данных, последовательность и вложенность которых определяет структуру документа и его содержание.

**Корневой тег** – тег с которого начинается документ.

атрибутная форма XML-документа - каждый XML-атрибут преобразовывается в столбец таблицы

элементная форма XML-документа - каждый элемент преобразовывается в столбец таблицы

1. Поясните понятия «XML-тип», «типизированные XML-данные», «коллекция XML-схем».

**XML-тип** – тип, который сообщает БД, что это XML.

**Типизированные XML-данные** – данные, которым соответствует XML-схема.

**Коллекция XML-схем** хранит импортированные XML-схемы и используется для решения следующих задач:

* проверка экземпляров XML;
* типизация XML-данных, хранимых в базе данных.