

Онлайн образование

otus.ru



Проверить, идет ли запись

Меня хорошо видно && слышно?



Тема вебинара

Основные компоненты и архитектура SQL Server

Курс “MS SQL Server Developer”

Разработчик MS SQL Server



Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в telegram



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения



Индивидуально



Время, необходимое
на активность



Пишем в чат



Говорим голосом



Документ



Ответьте себе или
задайте вопрос

Маршрут вебинара

Оргвопросы

Возможности SQL Server

Компоненты, архитектура SQL Server

Демо БД — WideWorldImporters

Домашнее задание



Цели вебинара

После занятия вы сможете

1. Разобраться с тем, как работать на курсе
 2. Посмотреть инструменты для работы с курсом
 3. Обсудить типы СУБД
-

Смысл

Зачем вам это уметь?

1. Чтобы понимать как работать на курсе и варианты работы с БД
-

О себе



Кристина Кучерова

Solution architect

Расскажите немного о себе:

- высшее математик-программист ЮРГТУ (НПИ)*
- с 2006 года работаю в ИТ*
- специализируюсь на БД и архитектуре систем*
- преподаватель курса "SQL Server разработчик",
"Практический курс по миграции с MS SQL Server на PostgreSQL" в ОТУС*



Команда курса



Марина Васильева



Дмитрий Тарасов



Евгений
Степанов



Кристина Кучерова



Людмила
Громницкая



Михаил ржевский



Познакомимся с вами

Напишите в чат уровень знакомства с SQL Server

0 — не знаком

1 — слабо знаком, писал простые SELECT

2 — уверенные знания

3 — перехожу с другой СУБД

Какую версию SQL Server используете

Пример:

2 2016

Организационные вопросы

Курс

- [Программа на сайте Otus](#)
- с 28.03.2024 — 02.10.2024 (последнее занятие 29.08.2024)
- 2 занятия в неделю по 2 академических часа (1,5 обычных часа)
 - понедельник 20:00 (мск)
 - четверг 20:00 (мск)
- Материалы в личном кабинете (ЛК)
- Записи занятий выкладываются в течение суток
- Каникулы (03.06 — 13.03, 15.07 — 25.07), по праздникам занятий нет
- Общение в чате вебинара, в телеграмме, личном кабинете

Домашнее задание

- ДЗ сдаем в виде **sql-скриптов**, текста в github.
Не скриншоты запросов. Скриншоты, если явно указано в ДЗ.
- Один репозиторий на весь курс (otus-mssql-<ваше_имя>)
- **Для проверки ДЗ ссылку на github в чат по ДЗ в ЛК** (кнопка "Отправить ДЗ на проверку") Подробнее про "Инструкция ЛК студента.pdf".
- Вопросы по **вашему ДЗ в ЛК**, общие вопросы по ДЗ лучше в телеграмм
- Время проверки ДЗ — **2 суток**. Если в течение этого времени нет ответа, то нужно писать в чате в телеграмм. Если сдаете "долги", то может быть дольше.
- Дедлайнов в течении курса нет. Дедлайн — **02.10.2024** окончание курса. В ЛК **рекомендуемые** сроки сдачи ДЗ.
- Сертификат выдается 02.10.2024. В сертификате будет указано сколько сдано ДЗ и название проекта.
- Проект.

Инструменты

Основные инструменты

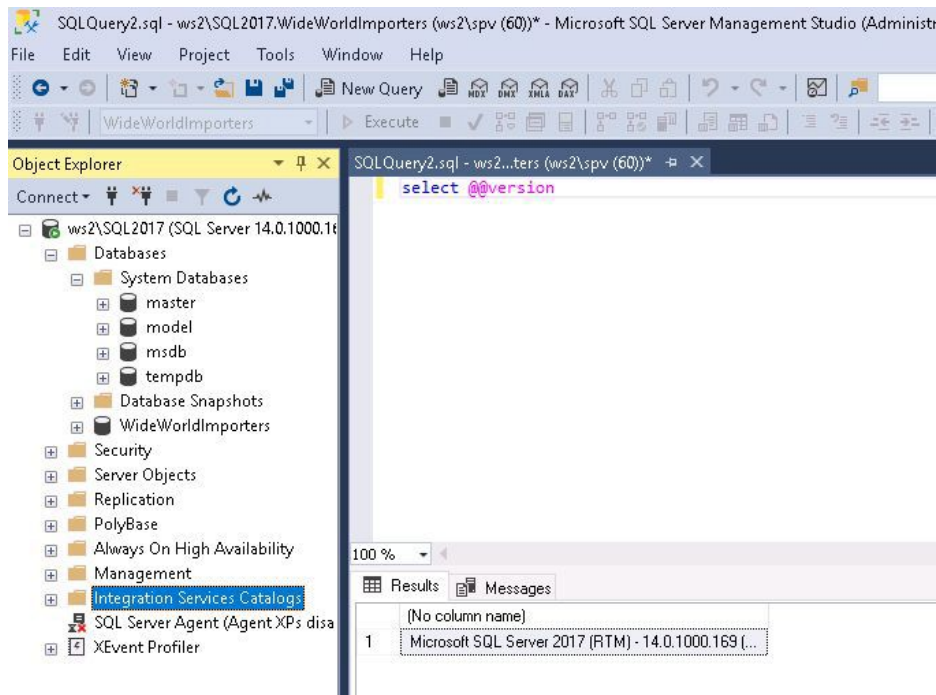
Основное:

- SQL Server 2017/2019/2022 Developer
- SQL Server Management Studio (SSMS)
- Visual Studio
- Git / Github
- Демо БД – WorldWideImporters

Дополнительно

- [SQL Sentry Plan Explorer](#)
- Azure Data Studio (Visual Studio Code)
- [SentryOne Plan Explorer Extension for Azure Data Studio](#)

SQL Server Management Studio

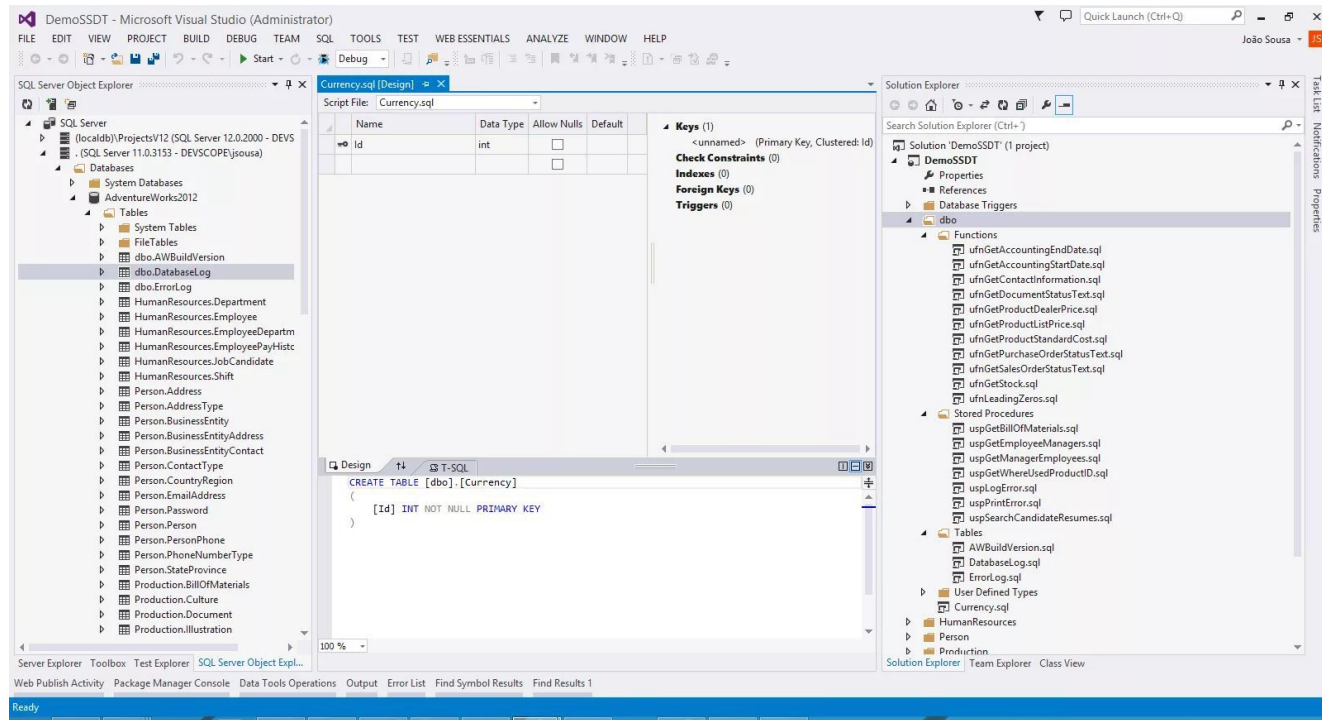


<https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver16>

Visual Studio Community

Нужен для
занятий

- CLR (C#)
- OLAP
- SSRS
- SSIS



<https://visualstudio.microsoft.com/ru/vs/community/>



Azure Data Studio

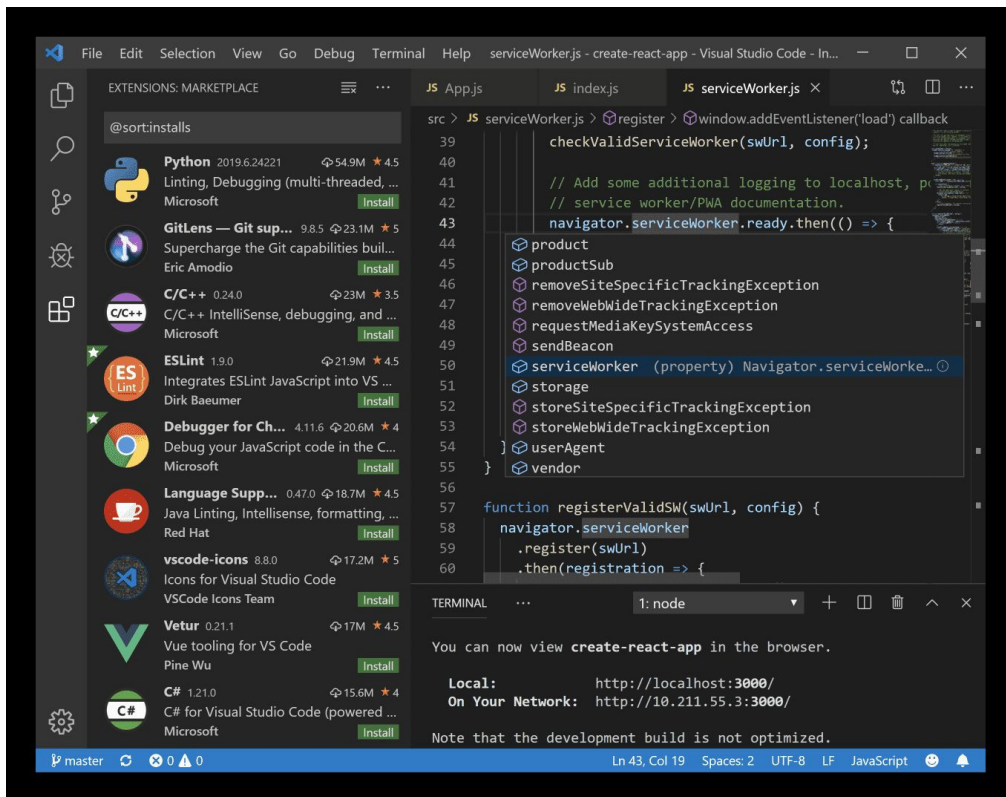
The screenshot displays the Azure Data Studio interface with the following components:

- Left Panel (Servers):** Shows a tree view of server groups. Under 'Development Server Group', 'WideWorldImporters' is selected.
- Top Panel (Query Editor):** Displays a SQL query named 'QDS_Query_Perf.sql'. The query is:

```
1 DECLARE @qds_status int = (SELECT actual_state FROM sys.database_query_store_options
2 IF @qds_status > 0
3 BEGIN
4 WITH SlowestQry AS(
5     SELECT TOP 5
```
- Results Panel:** Shows a table of query results with columns: query_id, last_execution_time, max_duration, and plan_id. The first row is highlighted.
- Right Panel (Visualizations):** Contains several charts:
 - QDS: Top 5 slowest queries:** A line chart showing max_duration over time for queries 41996, 42076, 42401, 42563, and 42569.
 - Space used by tables:** A horizontal bar chart comparing space usage (rows_count, total_pages, used_pages, data_pages, total_space_MB, used_space_MB, data_space_MB) for tables like Invoices, InvoiceLines, OrderLines, and CustomerTransactions.
 - Missing index and PKs:** A section for identifying missing indexes and primary keys.
 - Disk Usage:** A chart showing Used Space and Available Space.
 - Data file space usage (MB):** A chart showing reserved, data, index, and unused space.
- Bottom Panel (Status Bar):** Displays connection details: 'kubernetes : WideWorldImporters Ln 1, Col 1 Spaces: 4 UTF-8 LF SQL MSSQL'.

<https://learn.microsoft.com/en-us/sql/azure-data-studio/what-is-azure-data-studio?view=sql-server-ver16>

Visual Studio Code (+ mssql ext)



<https://code.visualstudio.com/>

GIT

Напишите в чат ваш уровень знакомства GIT / GitHub:

0 — не знаком совсем

1 — слышал, читал, что-то пробовал, но в работе не использую

2 — использую каждый день

3 — гуру SCM, CI, git, github

Материалы по GIT

<https://git-scm.com>

<https://tortoisegit.org>

Материалы

- <https://githowto.com/ru>
- <http://www-cs-students.stanford.edu/~blynn/gitmagic/intl/ru/>
- <https://git-scm.com/book/ru/v2>
- <https://guides.github.com/activities/hello-world/>



Пример структуры репозитория для ДЗ



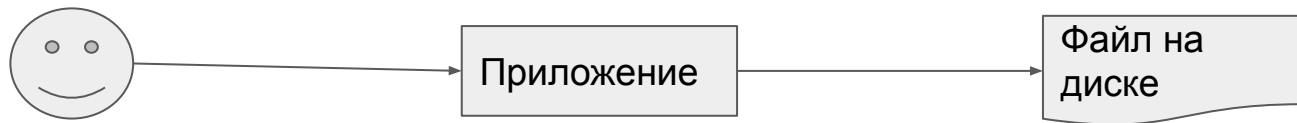
Каждое ДЗ в отдельной папке в виде sql-скриптов и др файлов. Папки лучше создавать по мере выполнения ДЗ, а не заранее.

Материалы по проекту в отдельной папке

Домашние задания курса OTUS "MS SQL Server разработчик".
Группа 2023-10.

Типы СУБД

Можем работать и без СУБД



Успей купить!



3 191 Р 5 790 Р
Бестселлер
Электробритва Philips S1332/41, черный



1 490 Р 2 180 Р
Бестселлер
Тонометр B.Well PRO-33 + адаптер + конусная...



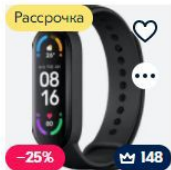
10 999 Р 12 999 Р
Бестселлер
Аппарат магнитотерапевтический...



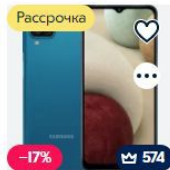
3 180 Р 4 990 Р
Бестселлер
Утюг Tefal Express Steam FV2866E0,...



8 999 Р 14 990 Р
Бестселлер
Бытовой пылесос Philips FC9332...



2 990 Р 3 990 Р
499 Р х 6 мес
Бестселлер
Фитнес-браслет Xiaomi Mi Smart Band 6,...



11 491 Р 13 980 Р
1 916 Р х 6 мес
Бестселлер
Смартфон Samsung Galaxy A12 4/64GB,...

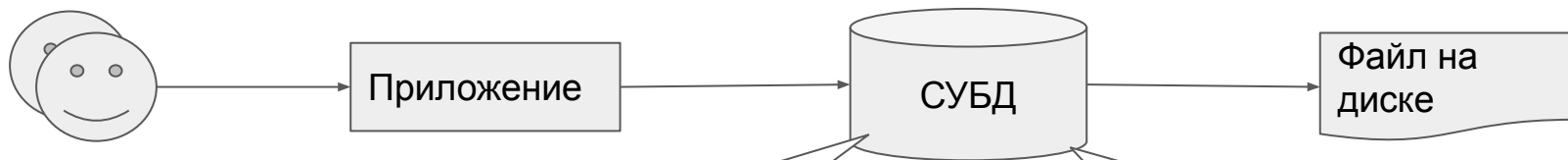


44 999 Р 49 980 Р
7 500 Р х 6 мес
Бестселлер
Смартфон Xiaomi 11T 8/256GB, серый + Mi...

Можем просто хранить их в файле (бинарный, текстовый, XML, JSON и др) и обращаться к ним напрямую из приложения.

Код товара;Артикул;Наименование;Цена;Закупочная цена;Количество;Размер;Цвет;Фото
1647;1647;Рубашка;790,00;0,00;1;;;0,00;0x0x0;vash-tovar;Каталог >> Категория 1;;
1648;1648-0;Часы;890,00;0,00;1;;;0,00;0x0x0;vash-tovar-1;Каталог >> Категория 1;
1649;1649;Платье;990,00;0,00;1;;;0,00;0x0x0;vash-tovar-2;Каталог >> Категория 1
1650;1650;Шорты;790,00;0,00;1;;;0,00;0x0x0;vash-tovar-3;Каталог >> Категория 2;K
1651;1651;Туфли;890,00;0,00;1;37;Голубой;;0,00;0x0x0;vash-tovar-4;Каталог >> Кате
1651;1651-2;;800,00;0,00;1;37;Белый;4486.jpg;0,00;0x0x0;vash-tovar-5;Каталог >> Кате
1651;1651-3;;800,00;0,00;1;38;Белый;4486.jpg;0,00;0x0x0;vash-tovar-5;Каталог >> Кате
1652;1652;Кепка;990,00;0,00;1;;;0,00;0x0x0;vash-tovar-5;Каталог >> Категория 2;;

Зачем нужна база данных (БД)?



А можем через СУБД.
БД — база данных
СУБД — система управления базами данных

Зачем нужна СУБД?
Почему приложению не работать с файлами с данными напрямую?
Напишите в чат.

Функции СУБД

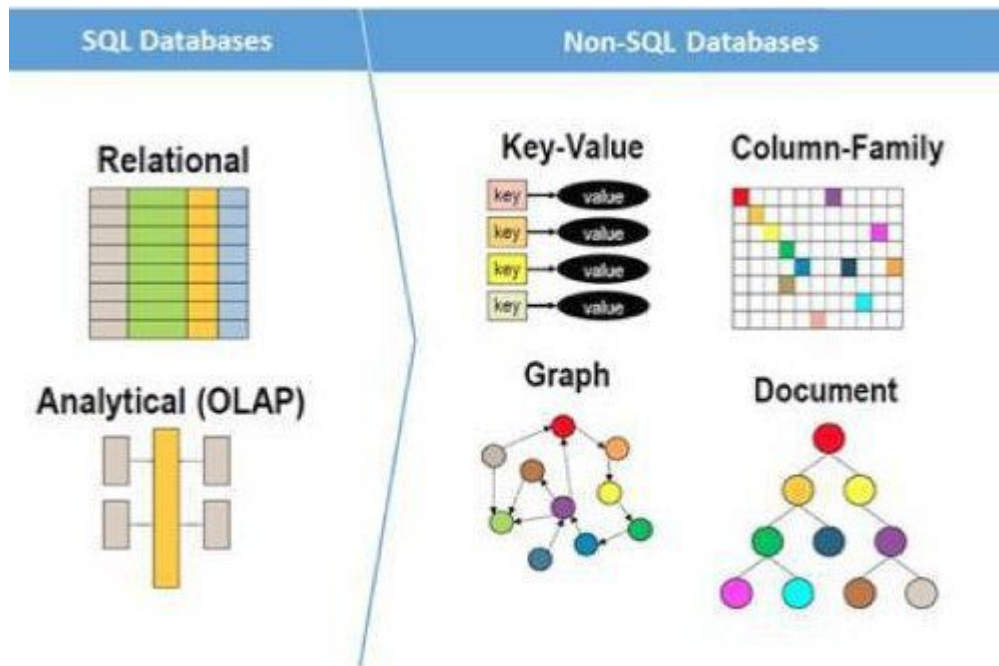
- Управление данными (SQL - SELECT, INSERT, UPDATE и т.п.)
- Целостность
- Конкурентный доступ (многопользовательская работа)
- Защита данных (разграничение прав доступа)
- Журнализация изменений, резервное копирование и восстановление базы данных после сбоев
- И другое

Какие типы БД (модели данных) вы знаете?

- К какому типу относится SQL Server?

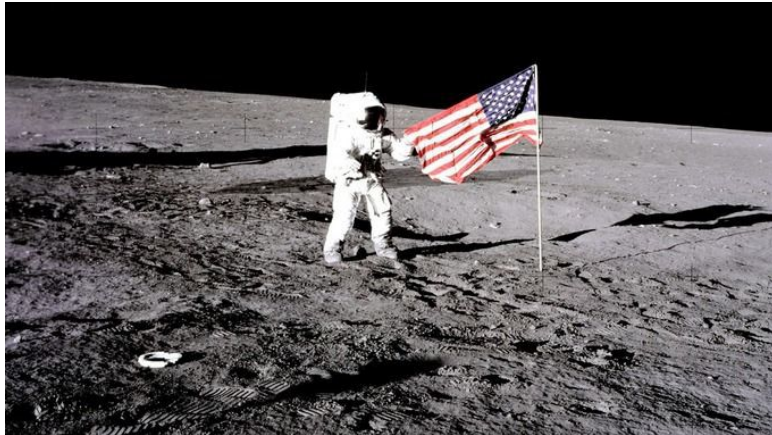
Какие БД бывают и зачем они используются?

1. Иерархические
2. Реляционные
3. Объектно-ориентированные
4. Документные
5. Графовые
6. In-Memory
7. Колоночные



Иерархические СУБД

1. IMS создана для «лунной» программы, чтобы посчитать сколько и куда уходят деньги
2. 1966 год
3. До сих пор используется 0,04% рынка СУБД
4. Порядка 90% компания из top Fortune
5. IMS обрабатывает порядка 50 миллиардов транзакций в день



Пример иерархической БД



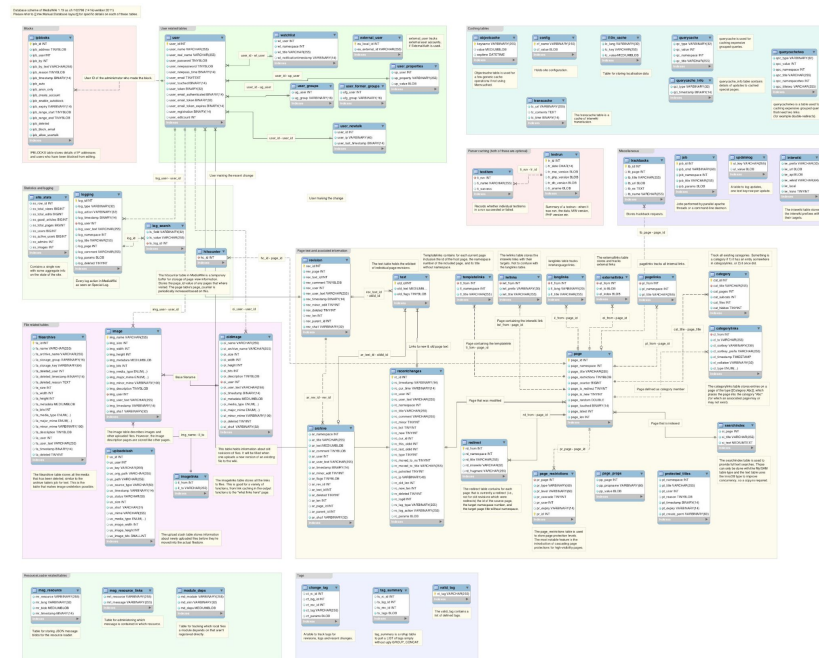
2015

© Богдан Н.А.

MyShared

Реляционные СУБД

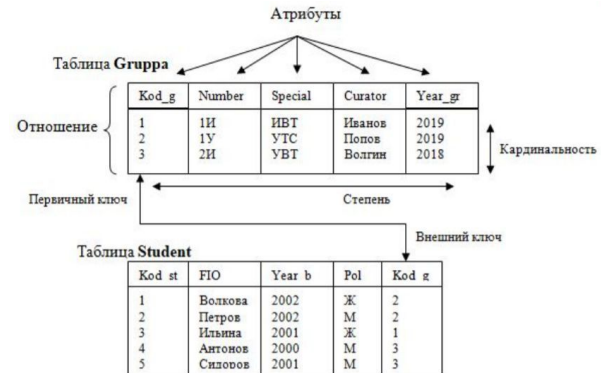
- Oracle, MS SQL Server, PostgreSQL, MySQL
- Основана на теории множеств, реляционная алгебра
- По простому - есть таблицы и связи между таблицами
- Эдгар Франк Кодд, 1970
 - [Е.Ф. Кодд. Реляционная модель данных для больших совместноиспользуемых банков данных](#)
 - [12 правил Кодда для РСУБД](#)
- Нормализация
- IBM System R - 1972
- Стандарт SQL - 1986



Реляционная модель

Формальный реляционный термин	Неформальный эквивалент
Отношение (relation)	Таблица
Кортеж (tuple)	Строка или запись
Кардинальность	Количество строк
Степень (отношения)	Количество столбцов
Первичный ключ	Уникальный идентификатор
Домен	Общая совокупность допустимых значений, похоже на "тип данных", но не совсем

- [Про "отношение" в wikipedia](#)
- Данные представлены в виде кортежей (tuple) == строки
- Объединенных в отношения (relation) == таблица
- Сначала необходимо описать "схему БД" - структуру таблиц (имя, имена колонок, типы колонок)

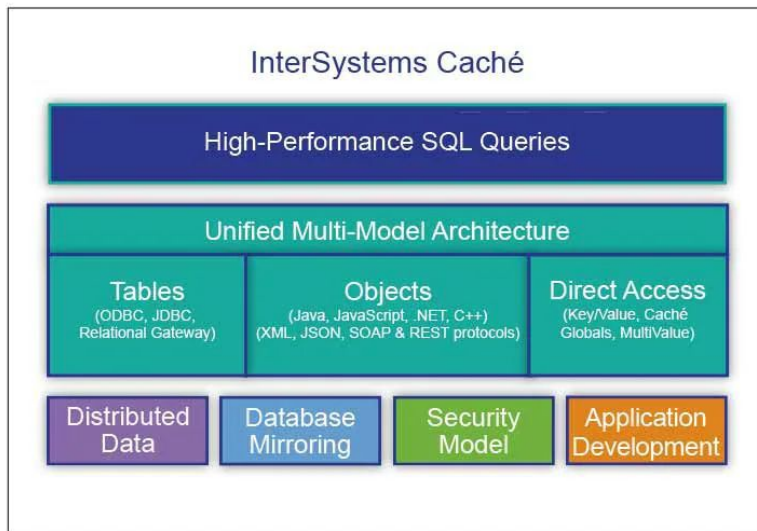


Основные понятия:

отношение;
атрибут;
тип данных;
домен;
кортеж;
первичный ключ;
внешний ключ.

Объектно-ориентированные

InterSystems Caché / IRIS



```
Class Demo.Person Extends (%Persistent, %Populate)
{
    Property Name As %String;
    Property Gender As %String;
    Method Print() As %Status
    {
        write !, ..Name, " is a ", ..Gender
    }
    ClassMethod ShowIDs() As %Status
    {
        &sql(declare ID cursor for select * into :person() from Demo.Person)
        &sql(open ID)
        write !, "Available IDs"
        write !, "-----"
        for {

            &sql(fetch ID)
            quit:(SQLCODE '= 0)
            write !, person(1), ?5, "- ", person(3)

        }
        &sql(close ID)
    }
}

<Storage name="Default">
</Storage>
}
```

The diagram illustrates the structure of a Caché class definition. It shows a class `Class Demo.Person` extending `(%Persistent, %Populate)`. The class contains three members: a `Property Name As %String;`, a `Property Gender As %String;`, and a `Method Print() As %Status`. The `Method Print()` contains a `write !, ..Name, " is a ", ..Gender` statement. Below the `Method Print()` is a `ClassMethod ShowIDs() As %Status`. The `ClassMethod ShowIDs()` contains a series of SQL statements: `&sql(declare ID cursor for select * into :person() from Demo.Person)`, `&sql(open ID)`, `write !, "Available IDs"`, `write !, "-----"`, a `for {` loop containing `&sql(fetch ID)`, `quit:(SQLCODE '= 0)`, and `write !, person(1), ?5, "- ", person(3)`, followed by `}`, `&sql(close ID)`, and `}`. At the bottom, there is a `<Storage name="Default">` tag followed by `</Storage>` and a closing brace `}`.

Annotations on the right side of the code block:

- Class Property** points to `Property Name As %String;`
- Instance Method** points to `Method Print() As %Status`
- Class Method** points to `ClassMethod ShowIDs() As %Status`

Документные

Документная база данных – это тип нереляционных баз данных, предназначенный для хранения и запроса данных в виде документов в формате, подобном JSON.
Пример СУБД: MongoDB

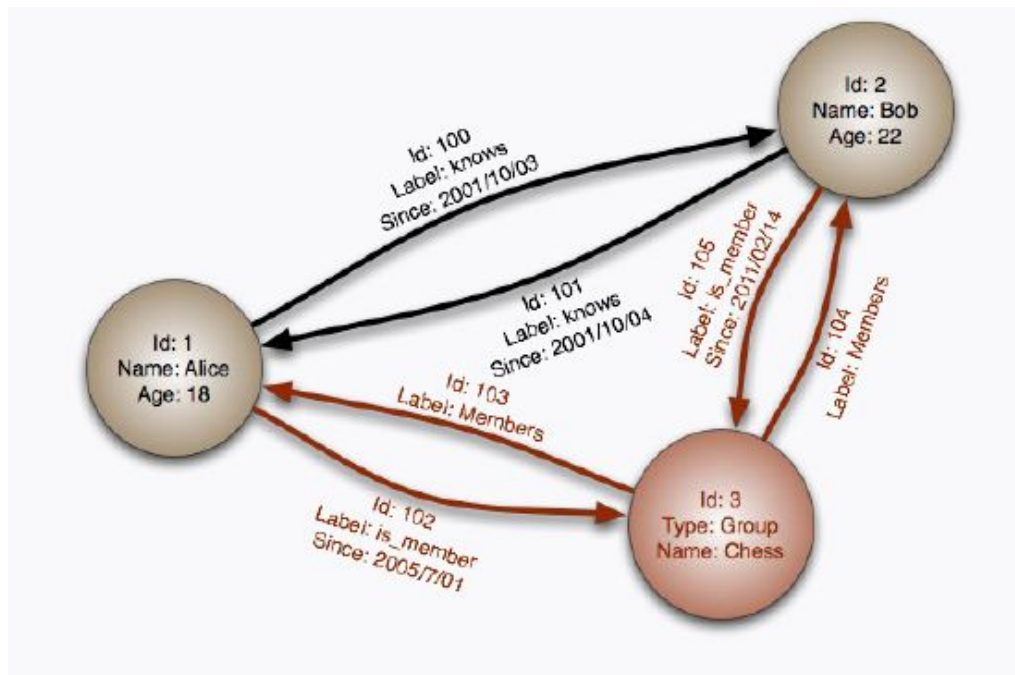
```
{
  "_id" : ObjectId("591f49970f7b726a5beb58af"),
  "name" : "Anna",
  "surname" : "Tudor",
  "music" : [
    "country",
    "blues",
    "chill-out"
  ],
  "age" : 18,
  "virgin" : true
}
```

```
{
  "_id" : ObjectId("591f49970f7b726a5beb58af"),
  "name" : "Anna",
  "surname" : "Tudor",
  "music" : [
    "country",
    "blues",
    "chill-out"
  ],
  "age" : 18,
  "virgin" : true
}
{
  "_id" : ObjectId("591f4eaf02ed4d6e1d81bc97"),
  "name" : "Leanne Graham",
  "username" : "Bret",
  "email" : "Sincere@april.biz",
  "address" : {
    "street" : "Kulas Light",
    "suite" : "Apt. 556",
    "city" : "Gwenborough",
    "zipcode" : "92998-3874",
    "geo" : {
      "lat" : "-37.3159",
      "lng" : "81.1496"
    }
  },
  "phone" : "1-770-736-8031 x56442",
  "website" : "hildegard.org"
}
```

Графовые

В графовых базах данных используются узлы для хранения сущностей данных и ребра для хранения взаимосвязей между сущностями.

Пример СУБД: Neo4j



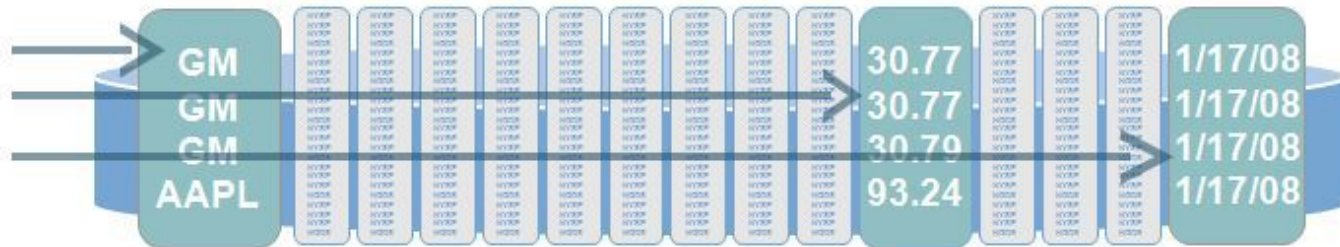
Колоночные

База данных, размещаемая в оперативной памяти

Пример СУБД: Vertica, ClickHouse

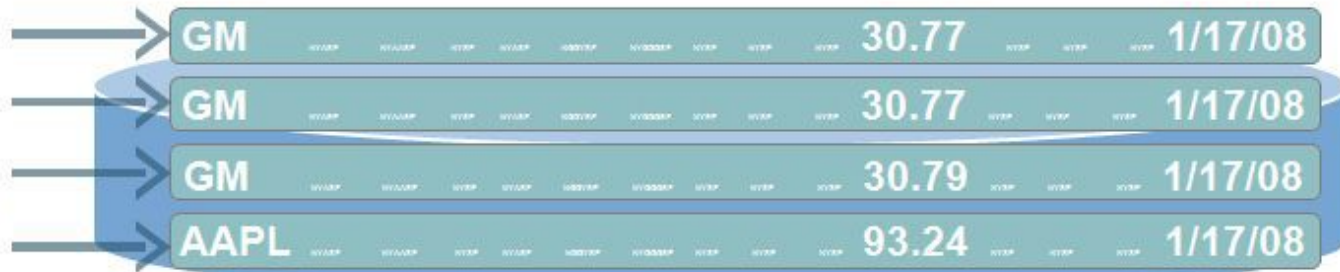
Хранение по столбцам

Чтение 3х столбцов



Хранение по строкам

Чтение всех столбцов



Что из этого относится к SQL Server?

1. Иерархические
2. Реляционные
3. Объектно-ориентированные
4. Документные
5. Графовые
6. In-Memory
7. Колоночные

Что из этого относится к SQL Server?

1. Иерархические Hierarchical Data — hierarchyid

```
USE AdventureWorks ;
GO

CREATE TABLE Org_T1
(
    EmployeeId hierarchyid PRIMARY KEY,
    OrgLevel AS EmployeeId.GetLevel(),
    EmployeeName nvarchar(50)
);
GO

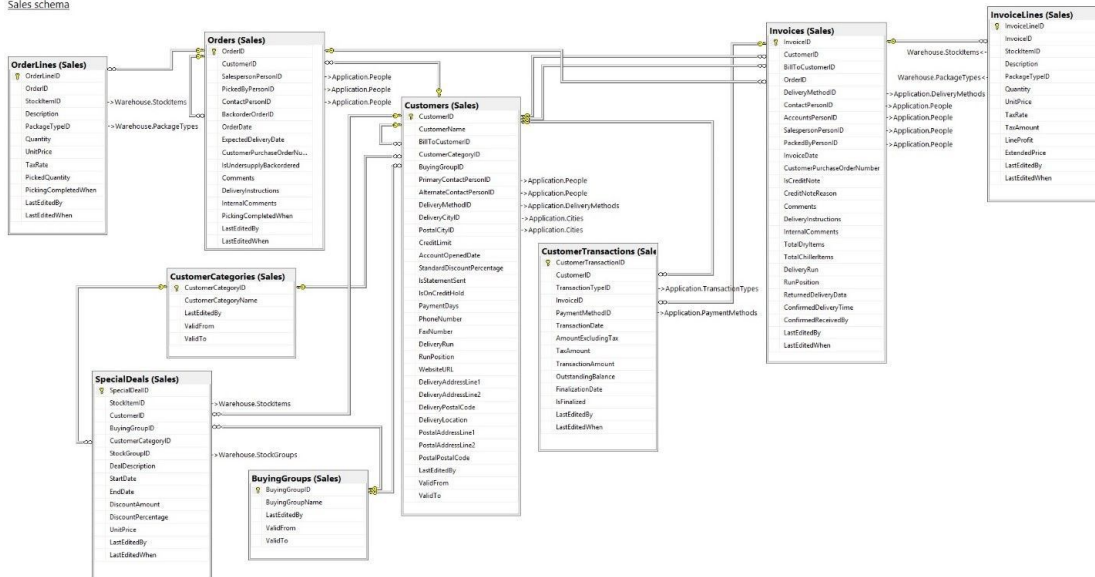
CREATE INDEX Org_BreadthFirst ON Org_T1(OrgLevel, EmployeeId);
GO

CREATE PROCEDURE AddEmp(@mgrid hierarchyid, @EmpName nvarchar(50) )
AS
BEGIN
    DECLARE @last_child hierarchyid;
INS_EMP:
    SELECT @last_child = MAX(EmployeeId) FROM Org_T1
        WHERE EmployeeId.GetAncestor(1) = @mgrid;
    INSERT INTO Org_T1 (EmployeeId, EmployeeName)
        SELECT @mgrid.GetDescendant(@last_child, NULL), @EmpName;
    -- On error, return to INS_EMP to recompute @last_child
    IF @@error <> 0 GOTO INS_EMP
END ;
GO
```

Что из этого относится к SQL Server?

1. Иерархические Hierarchical Data — hierarchyid
2. Реляционные

Sales schema



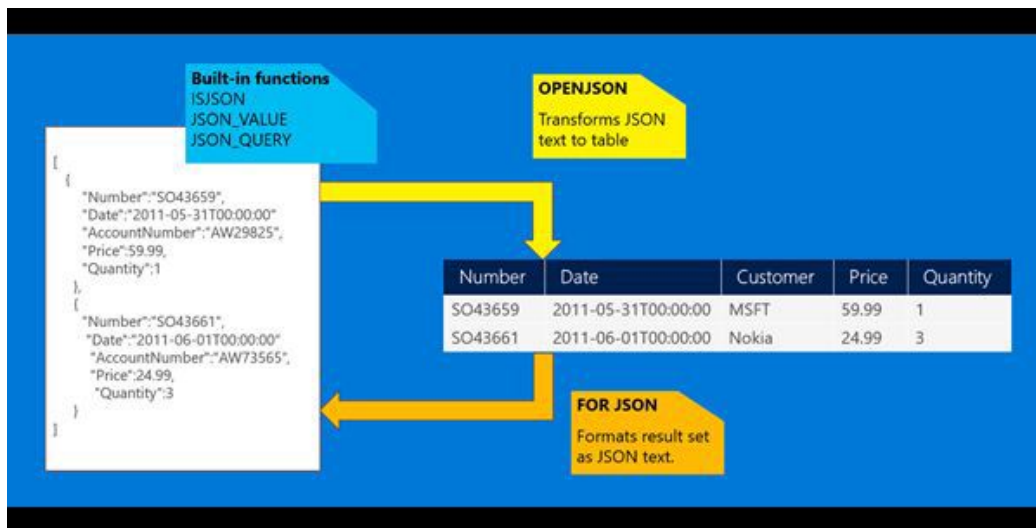
* All LastEditedBy columns reference ApplicationPeople

Что из этого относится к SQL Server?

1. Иерархические [Hierarchical Data](#) — hierarchyid
2. Реляционные
3. Объектно-ориентированные

Что из этого относится к SQL Server?

1. Иерархические [Hierarchical Data](#) — hierarchyid
2. Реляционные
3. Объектно-ориентированные
4. Документные [JSON](#) (с 2016), [XML](#) (с 2000)



Что из этого относится к SQL Server?

1. Иерархические [Hierarchical Data](#) — hierarchyid
2. Реляционные
3. Объектно-ориентированные
4. Документные [JSON](#) (с 2016), [XML](#) (с 2000)
5. Графовые

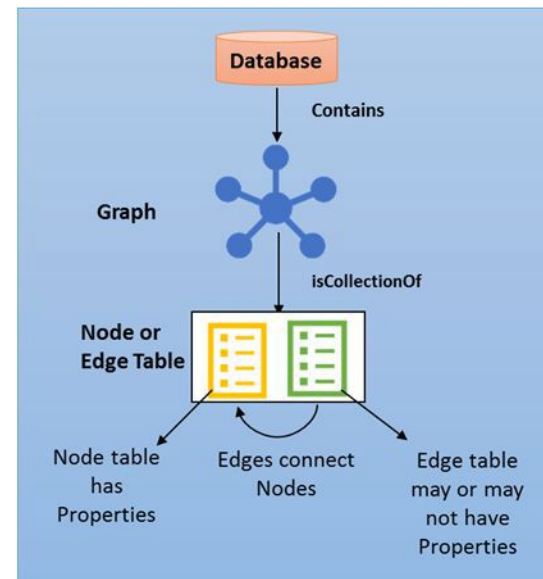
[Graph processing with SQL Server and Azure SQL Database](#)

[Введение в графовые базы данных SQL Server 2017](#)

Node Properties			Nodes that this edge connects				Edge Properties
\$node_id	Name	Age	\$edge_id	\$from_id	\$to_id	StartDate	
{"type": "node", "id": 0}	John	30	{"type": "edge", "id": 0}	{"type": "node", "id": 0}	{"type": "node", "id": 1}	01/01/2013	
"type": "node", "id": 1}	Mary	28	{"type": "edge", "id": 1}	{"type": "node", "id": 1}	{"type": "node", "id": 2}	05/05/2010	
"type": "node", "id": 2}	Alice	25	{"type": "edge", "id": 2}	{"type": "node", "id": 2}	{"type": "node", "id": 0}	09/09/2016	

Person Node Table

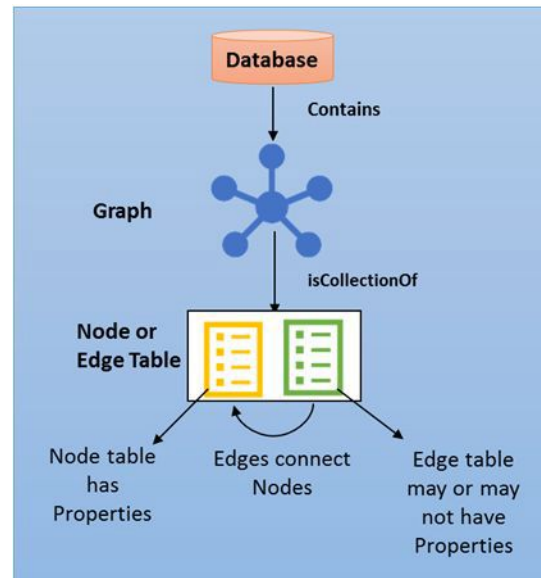
Friends Edge Table



Что из этого относится к SQL Server?

1. Иерархические [Hierarchical Data](#) — hierarchyid
2. Реляционные
3. Объектно-ориентированные
4. Документные [JSON](#) (с 2016), [XML](#) (с 2000)
5. Графовые
[Graph processing with SQL Server and Azure SQL Database](#)
[Введение в графовые базы данных SQL Server 2017](#)
6. In-Memory [In-Memory OLTP](#)
7. Колоночные [Columnstore indexes](#)

А еще есть полнотекстовый поиск, очереди и многое другое...



MS SQL Server

Версии SQL Server

- 1.0 — 1989 (Sybase, IBM OS/2), 4.2x — 1992-1993 (Windows NT), 6.x — 1995, 7.0 — 1998
- 2000 (8.0) — functions, indexed views, INSTEAD OF и AFTER триггеры, XML и др
- 2005 (9.0) — TRY/CATCH, PIVOT/UNPIVOT, CLR, Snapshot Isolation, Service Broker, mirror и др
- 2008 (10.0) — TDE, Extended Events, Resource Governor, Spatial Data Types
- 2008 R2 (10.5) — Master Data Services, StreamInsight, Multi Server Management, PowerPivot
- 2012 (11.0) — Always On, Sequence, Columnstore Indexes (readonly), Data Quality Services
- 2014 (12.0) — In-memory OLTP, Always On вторичная реплика для чтения, Delayed Durability



Версии SQL Server

- 2016 (13.0) — JSON, Temporal tables, Always Encrypted, Dynamic Data Masking, Row Level Security, нормальные Columnstore Indexes, QueryStore, Polybase, [много из Enterprise-фич стали доступны в Standard](#)
- 2017 (14.0) — Distributed Availability Group, Linux, Machine Learning (Python и R), SQL Graph
- 2019 (15.0) — Docker, репликация в Linux, Java и много других улучшений
- 2022 (16.0) — Много улучшений
<https://learn.microsoft.com/ru-RU/sql/sql-server/what-s-new-in-sql-server-2022?view=sql-server-ver16>

Что нового в SQL Server 2022 // Демо-занятие курса «MS SQL Server Developer»

<https://www.youtube.com/watch?v=XxYx9lt3k-Y>

Редакции MS SQL Server

- Standard
- Enterprise
- Web
- Express
- Developer

<https://learn.microsoft.com/en-us/sql/sql-server/editions-and-components-of-sql-server-2017?view=sql-server-ver16>

Установка

- Локальная установка ([ссылка на установщик SQL Server 2022](#))
 - В ЛК есть памятка по установке - “Настройка окружения.pdf”
- Виртуальная машина / vagrant ([ссылка на vagrant box](#))
 - Ставим VirtualBox, Vagrant, скачиваем Vagrantfile → vagrant up
- Linux
 - [Руководство по установке SQL Server на Linux](#)
- Docker
 - [Документация Microsoft по docker и SQL Server](#)
- Azure, AWS, GCP и др



Компоненты и архитектура

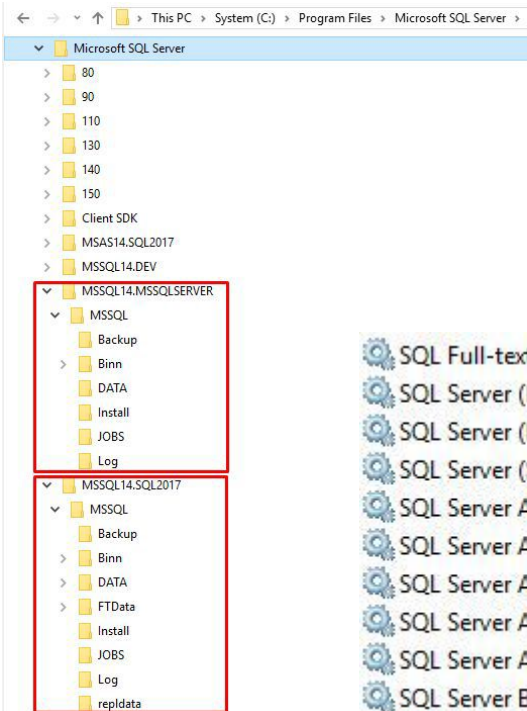
Компоненты SQL Server

- SQL Server (Database Engine)
 - SQL Server Replication
- SQL Server Agent
- SQL Browser
- Business Intelligence
 - SQL Server Integration Services (SSIS)
 - ETL - Extract, Transform, Load
 - SQL Server Analysis Services (SAS)
 - OLAP - Online Analytical Processing
- SQL Server Reporting Services (SSRS)
- SQL Full-Text Deamon

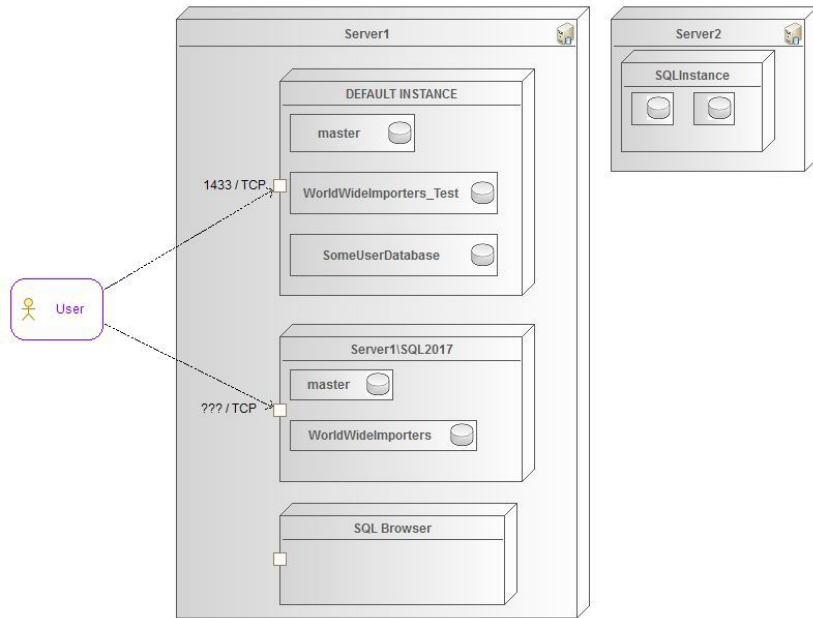


Экземпляры (instance)

Подключение к инстансу:
SQL_SERVER \ INSTANCE_NAME



- SQL Full-text Filter Daemon Launcher (SQL2017)
- SQL Server (DEV)
- SQL Server (MSSQLSERVER)
- SQL Server (SQL2017)
- SQL Server Agent (DEV)
- SQL Server Agent (MSSQLSERVER)
- SQL Server Agent (SQL2017)
- SQL Server Analysis Services (SQL2017)
- SQL Server Analysis Services CEIP (SQL2017)
- SQL Server Browser



Назовите имя сервера, имя экземпляра

1. SQL01
2. SQL01\SQL2017
3. .\dev
4. SQL02,12345
5. ADMIN:SQL02\TEST

Назовите имя сервера, имя экземпляра

1. SQL01
Сервер: SQL01, экземпляр - по умолчанию (служба MSSQLSERVER)
2. SQL01\SQL2017
Сервер: SQL01, экземпляр - SQL2017 (служба MSSQL\$SQL2017)
3. .\dev
Сервер: localhost, экземпляр - dev
4. SQL02,12345
Сервер: SQL02, экземпляр - не известно, т.к. Подключение по порту, порт 12345
5. ADMIN:SQL02\TEST
Выделенное административное подключение, Dedicated Administrator Connection (DAC)
<https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/diagnostic-connection-for-database-administrators>
Сервер: SQL02, экземпляр - TEST

SQL Server Configuration Manager

The image shows the SQL Server Configuration Manager interface with two dialog boxes open to configure TCP/IP properties.

SQL Server Configuration Manager (Local)

- SQL Server Services
- SQL Server Network Configuration (32bit)
- SQL Native Client 11.0 Configuration (32bit)
- SQL Server Network Configuration**
 - Protocols for SQL2017
 - Protocols for MSSQLSERVER
 - Protocols for DEV
- SQL Native Client 11.0 Configuration

Protocol Name Status

Protocol Name	Status
Shared Memory	Enabled
Named Pipes	Disabled
TCP/IP	Disabled

TCP/IP Properties (Top Dialog)

Protocol IP Addresses

General

Property	Value
Enabled	Yes
Keep Alive	30000
Listen All	Yes

TCP/IP Properties (Bottom Dialog)

Protocol IP Addresses

IP3

Property	Value
Active	Yes
Enabled	No
IP Address	:::1
TCP Dynamic Ports	0
TCP Port	

IP4

Property	Value
Active	Yes
Enabled	No
IP Address	127.0.0.1
TCP Dynamic Ports	0
TCP Port	

IPAll

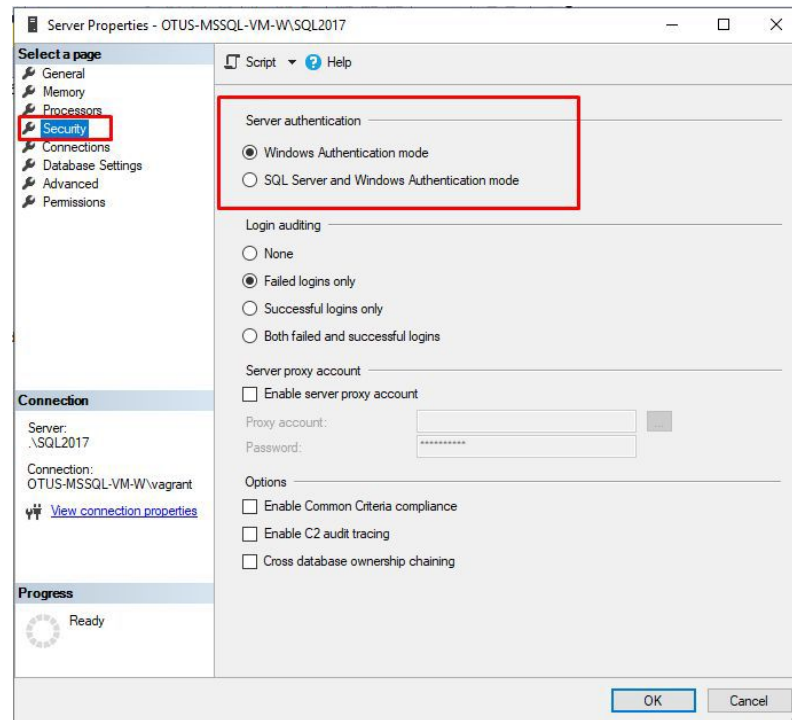
Property	Value
TCP Dynamic Ports	50022
TCP Port	

Active
Indicates whether the selected IP Address is active.

Buttons: OK, Cancel, Apply, Help

Аутентификация

- SQL Server authentication
- Windows authentication
 - Active Directory
- sa
- sysadmin



WideWorldImporters

<https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak>

Описание от Microsoft

- <https://learn.microsoft.com/ru-ru/sql/samples/wide-world-importers-what-is?view=sql-server-ver16>
- <https://learn.microsoft.com/ru-ru/sql/samples/wide-world-importers-oltp-database-catalog?view=sql-server-ver16>

LIVE

Развертывание WWI



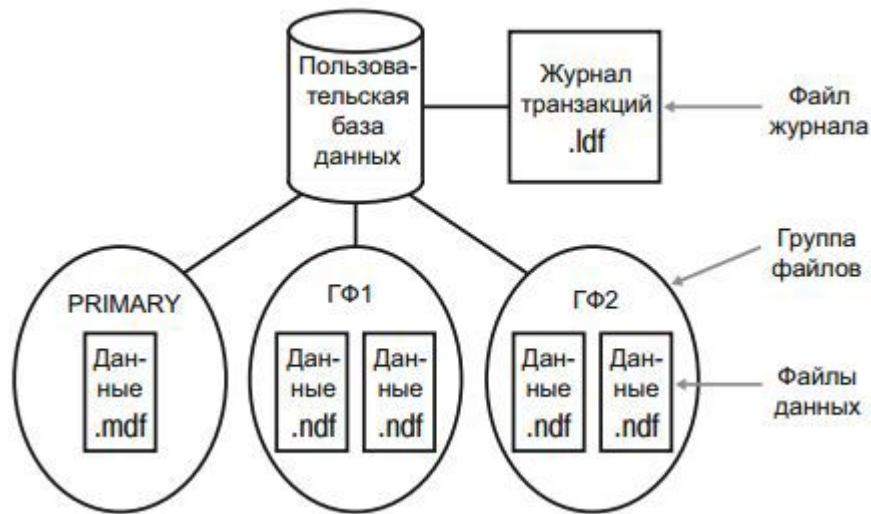
Системные БД

- master
 - Системная информация
- model
 - Шаблон для новых БД
- msdb
 - SQL Agent, jobs, backup
- tempdb
 - Временные объекты



Физическая структура

- Файловые группы (FileGroup)
 - Данные
 - Индексы
- Файлы на диске
 - Данные (*.mdf, *.ndf)
 - Журнал транзакций (*.ldf)

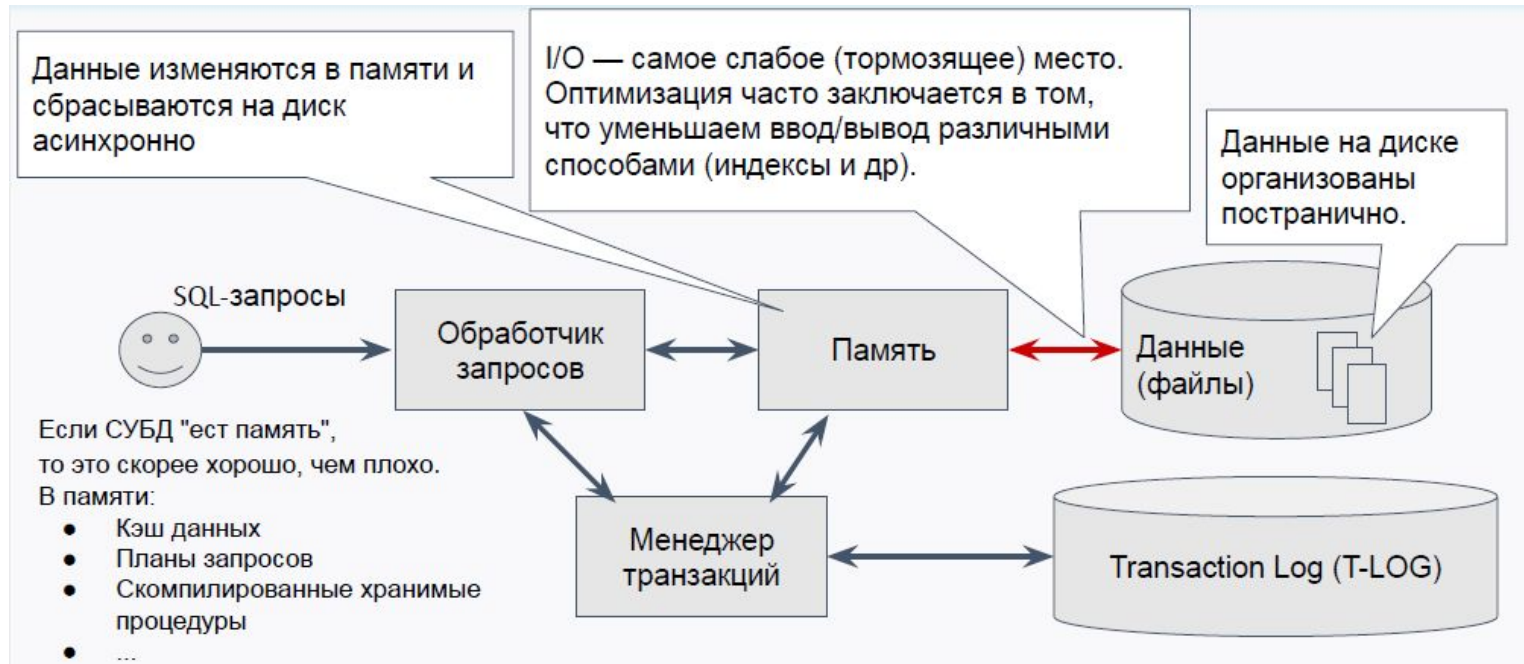


Database files:

Logical Name	File Type	Filegroup	Size (MB)	Autogrowth / Maxsize	Path	File Name
WWI_Primary	ROWS Data	PRIMARY	1024	By 64 MB, Unlimi...	...	C:\... WideWorldImporters.mdf
WWI_UserData	ROWS Data	USERDATA	2048	By 64 MB, Unlimi...	...	C:\... WideWorldImporters_UserData.ndf
WWI_InMemory_Data_1	FILESTREAM Data	WWI_InMemory_Data	125	Unlimited	...	C:\...
WWI_Log	LOG	Not Applicable	100	By 64 MB, Limite...	...	C:\... WideWorldImporters.ldf

Таблица → Файловая группа → Физический файл на диске

Архитектура СУБД



Архитектура SQL Server

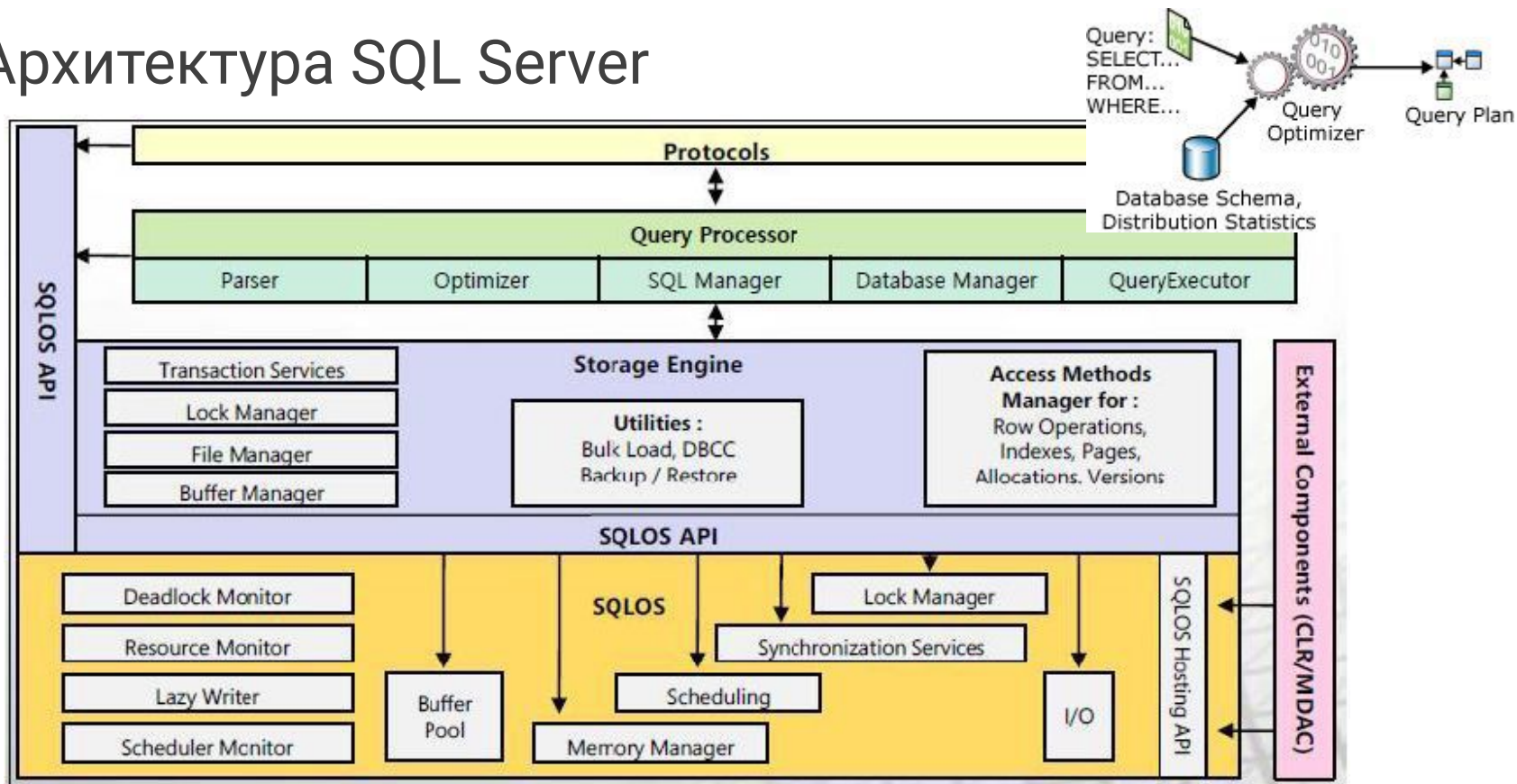
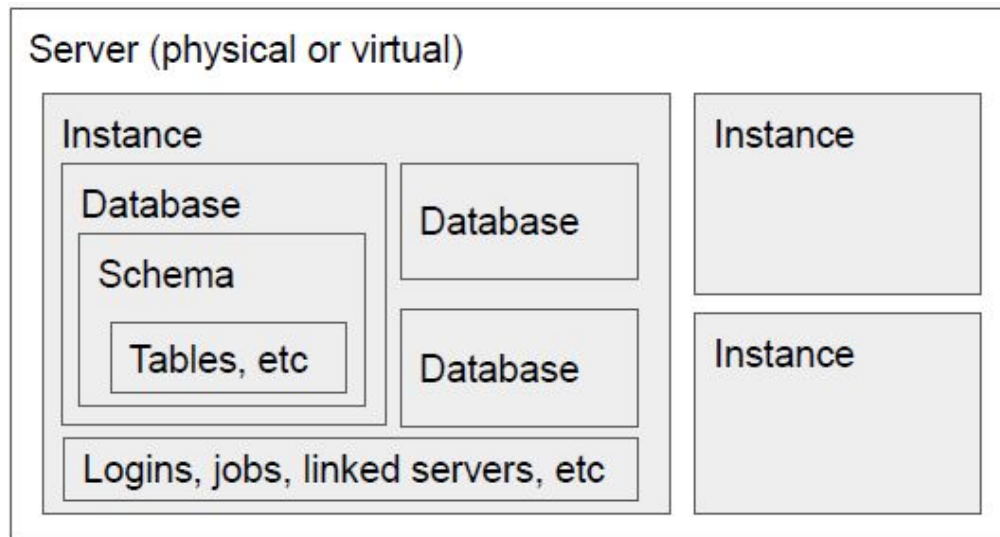


Figure 1. SQL Server architecture

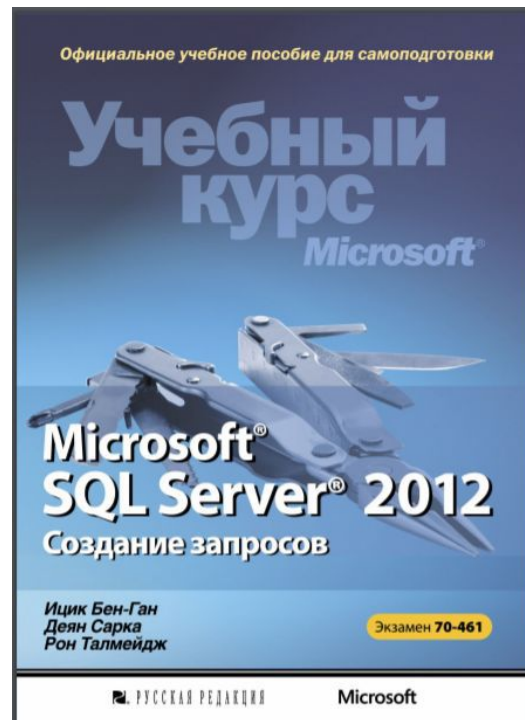
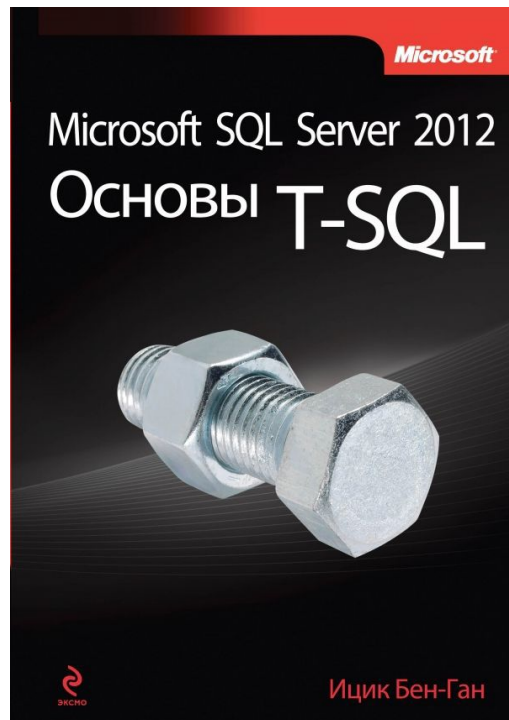
Компоненты SQL Server Database Engine

База данных (database)

- Схема (schema)
 - Таблица
 - Представление
 - Индексы
 - Хранимые процедуры
 - Функции
 - Ограничения
 - Синонимы
 - Типы
- Пользователи и роли
 - Job
 - Linked Server
 - Service Broker (очереди)
 - Репликация, Always On



Литература



Домашнее задание

Домашнее задание (обязательное)

1. Создайте github-репозиторий для хранения выполненных домашних заданий.
2. Установите SQL Server 2017\2019\2022 и SQL Server Management Studio.
3. Разверните у себя бэкап базы WideWorldImporters (из файла WideWorldImporters-Full.bak). Сделайте скриншот с восстановленной БД WideWorldImporters и выполненным `select @@version`. Скриншот разместите в вашем github-репозитории.
4. Прочитайте описание БД WideWorldImporters:
 - <https://learn.microsoft.com/en-us/sql/samples/wide-world-importers-what-is?view=sql-server-ver16>
 - <https://learn.microsoft.com/en-us/sql/samples/wide-world-importers-oltp-database-catalog?view=sql-server-ver16>

В ЛК памятка по выполнению ДЗ – “Настройка окружения.pdf”



Домашнее задание (опциональное)

6. Поставьте также SQL Sentry Plan Explorer и Visual Studio. Сейчас это необязательно, можно поставить позже, когда будут соответствующие темы.
7. В рамках курса предусмотрен проект. Работа над ним будет со второго модуля. Но если у вас уже есть идеи по проекту, то можете сделать его краткое описание.

В качестве проекта вы можете взять любую идею, которая вам близка. Это может быть какая-то часть вашего рабочего проекта, которую вы хотите переосмыслить. Если есть идея, но не понятно как ее уложить в рамки учебного проекта, напишите преподавателю и мы поможем. Проект мы будем делать весь курс и защищать его в самом конце, он будет заключаться в созданной БД со схемой, описанием проекта, и необходимыми процедурами\функциями или SQL кодом для демонстрации основного функционала системы.

Пока достаточно создать в github папку с проектом, создать там общее описание проекта - о чем он, какие функции будут реализованы, основные сущности (просто описание текстом). Активная работа над проектом будет со второго модуля.

Вопросы?



Ставим “+”,
если вопросы есть



Ставим “-”,
если вопросов нет

Рефлексия

Вопросы для проверки

1. Отметьте 3 пункта, которые вам запомнились с вебинара?

2. Что было нового?

Рефлексия



С какими основными мыслями
и инсайтами уходите с вебинара?



Как будете применять на практике то,
что узнали на вебинаре?



**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**