

Настройка окружения

В данном документе описана установка, настройка инструментов и окружения для прохождения курса **"Разработчик MS SQL Server"** в OTUS.

Содержание

Введение	2
1. Локальная установка SQL Server в Windows	3
2. SQL Server Configuration Manager	9
3. SQL Server Management Studio (SSMS)	10
4. SQL Sentry Plan Explorer	11
5. Visual Studio 2019 Community	12
6. БД WideWorldImporters	14
6.1. Разворачивание БД WideWorldImporters	14
6.2. Описание БД WideWorldImporters	17
7. Git / GitHub	18
7.1. Установка git в Windows	18
7.2. Рекомендации к оформлению репозитория	21
7.3. Ресурсы для изучения git / github	22

Введение

На курсе используются следующие основные инструменты:

- **SQL Server 2017 Developer Edition**
 - [Локальная установка SQL Server в Windows](#)
- **SQL Server Management Studio (SSMS)**
Инструмент для управления SQL Server и написания запросов.
- **Visual Studio 2019 Community**
Потребуется для занятий по CLR, OLAP, SSRS.
- **Git / GitHub**
Система контроля версий (для хранения выполненных ДЗ).

Для прохождения курса рекомендуется использовать Windows 10, хотя можно и Linux (в т.ч. Docker), но демонстрация на вебинарах и учебные материалы ориентированы на Windows. Также можно развернуть SQL Server в облаках (но в курсе это не рассматривается).



- Курс ориентирован на использование SQL Server 2017 Developer Edition в Windows.
- Best practice по установке SQL Server не рассматривается. Здесь приведена только минимальная информация для прохождения курса.
- По SQL Server в Linux и Docker можете посмотреть следующие открытые вебинары OTUS:
 - [Использование Microsoft SQL Server в Linux](#)
 - [SQL Server и Docker](#)

Вам также понадобится аккаунт на [github](#) для сдачи ДЗ. Можно использовать существующий аккаунт или можно создать новый специально для курса. Не обязательно github, можно любой другой хостинг git.

1. Локальная установка SQL Server в Windows

1. Ссылка для скачивания установщика SQL Server 2017 — <https://download.microsoft.com/download/5/A/7/5A7065A2-C81C-4A31-9972-8A31AC9388C1/SQLServer2017-SSEI-Dev.exe>
2. Выберите **Custom**.

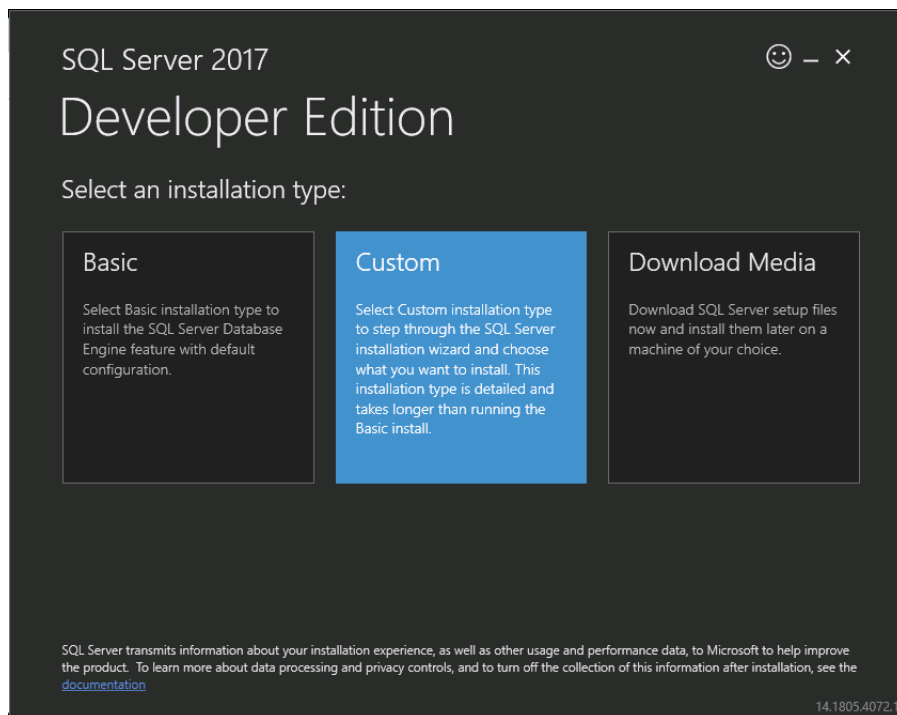


Рисунок 1. Выбор типа установки SQL Server

3. Укажите место для скачивания дистрибутива SQL Server.

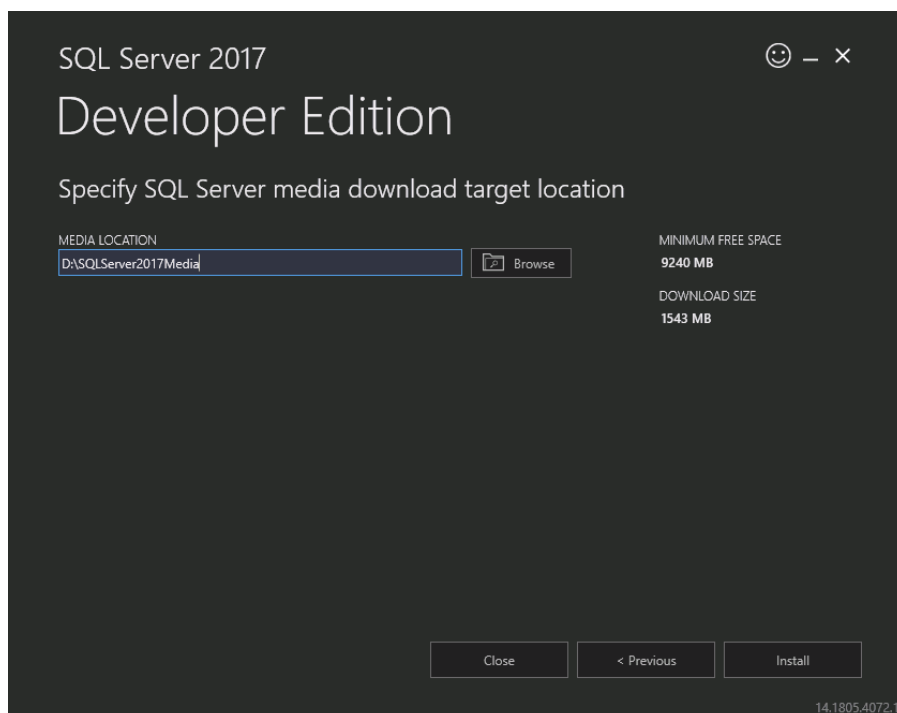


Рисунок 2. Указание места скачивания дистрибутива

4. Подождите, пока скачается дистрибутив.

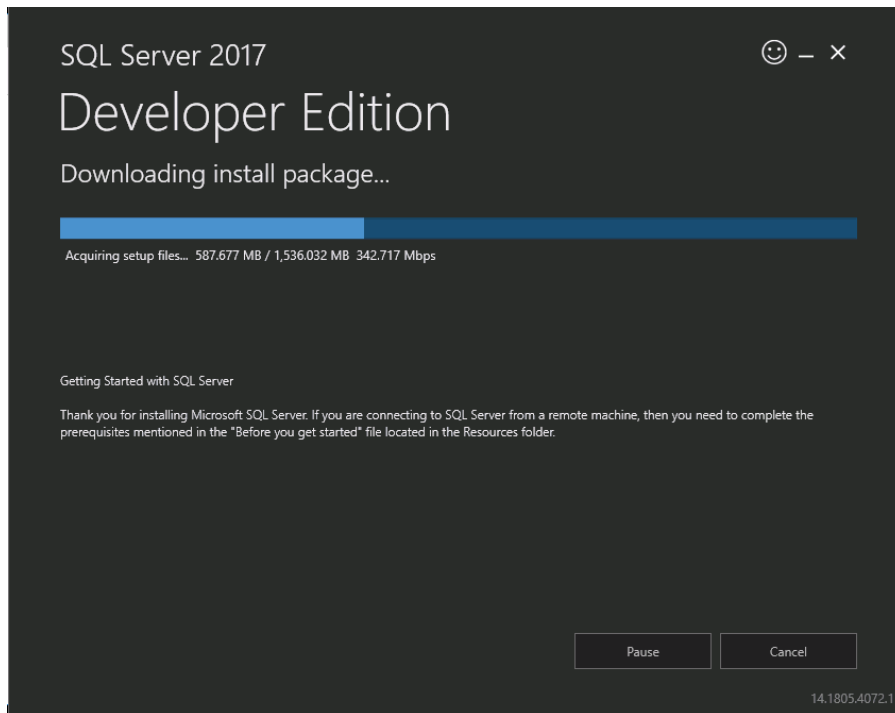


Рисунок 3. Скачивание дистрибутива SQL Server

5. Запустится SQL Server Installation Center.

Выберите **Installation** > **New SQL Server stand-alone installation or add features to an existing installation**.

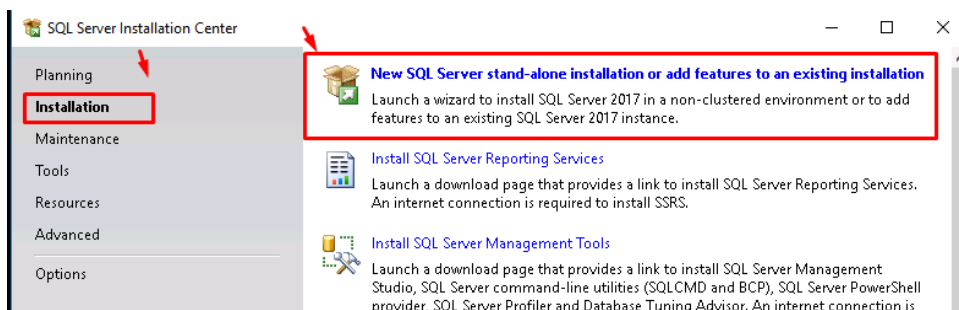


Рисунок 4. SQL Server Installation Center

6. Выбор редакции SQL Server. В поле **Specify Free Edition** выберите *Developer*.

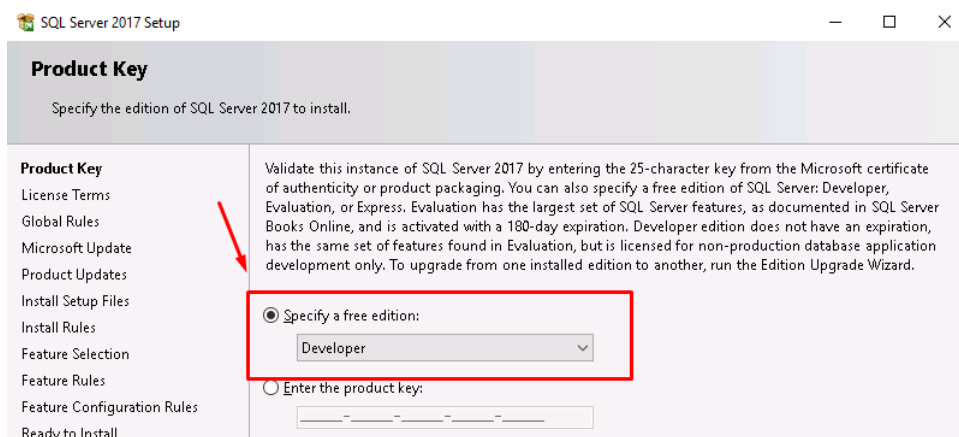


Рисунок 5. Выбор редакции SQL Server

7. Прокликайте [**Next**] до выбора компонент (**Feature Selection**).

8. Feature Selection.

Минимальный набор компонент для курса следующий (при желании можно установить все компоненты):

- Database Engine
 - SQL Server Replication
 - Full-Text and Semantic Extractions for Search
- Analysis Services
- Shared Features
 - Client Tools Connectivity

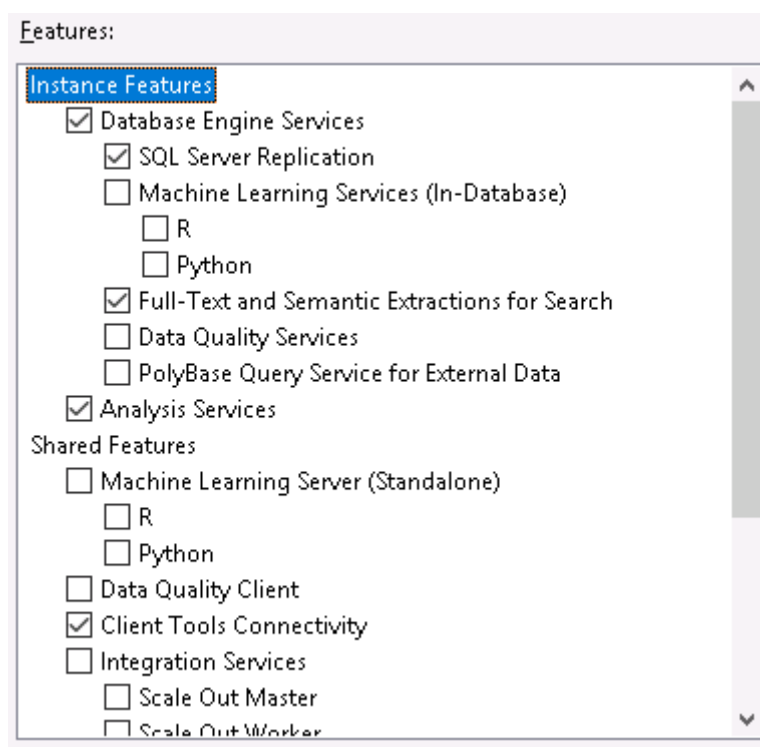


Рисунок 6. Выбор компонент

9. Выберите тип экземпляра (на свое усмотрение): по умолчанию (**Default instance**) или именованный (**Named Instance**). Для именованного укажите имя.

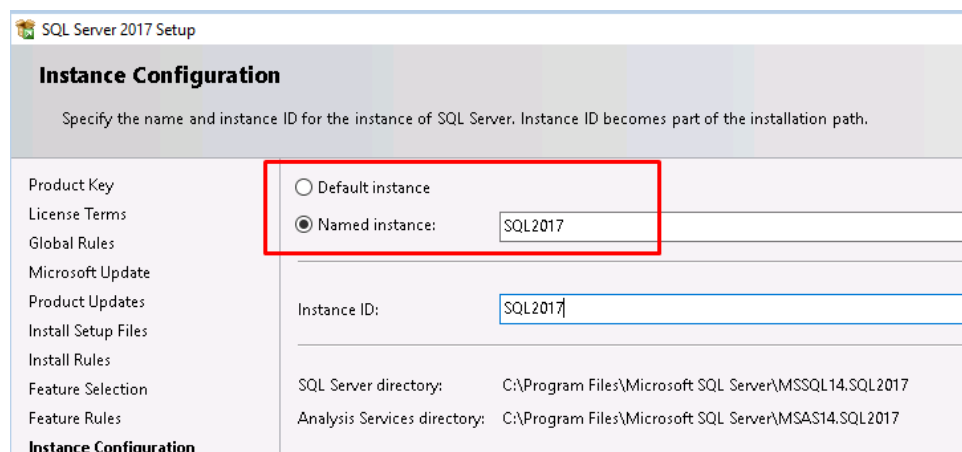


Рисунок 7. Имя экземпляра

10. Параметры запуска служб.

Service Accounts Collation

Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Password	Startup Type
SQL Server Agent	NT Service\SQLAgent\$SQL2017		Manual
SQL Server Database Engine	NT Service\MSSQL\$SQL2017		Automatic
SQL Server Analysis Services	NT Service\MSOLAP\$SQL2017		Automatic
SQL Full-text Filter Daemon Launcher	NT Service\MSSQLFDLauncher\$SQL20...		Manual
SQL Server Browser	NT AUTHORITY\LOCAL SERVICE		Automatic

☐ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

[Click here for details](#)

Рисунок 8. Конфигурация служб

Настройку **Collation** (параметры сортировки) оставьте по умолчанию.

Service Accounts Collation

Database Engine:

SQL_Latin1_General_CP1_CI_AS Customize...

Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 52 on Code Page 1252 for non-Unicode Data

Analysis Services:

Latin1_General_CI_AS Customize...

Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

Warning: The Database Engine and Analysis Services collations are not the same.

Рисунок 9. Настройка Collation

11. Параметры аутентификации.

Для предоставления текущему пользователю прав администратора SQL Server нажмите кнопку [**Add Current User**] .

Server Configuration Data Directories TempDB FILESTREAM

Specify the authentication mode and administrators for the Database Engine.

Authentication Mode

☒ Windows authentication mode

☐ Mixed Mode (SQL Server authentication and Windows authentication)

Specify the password for the SQL Server system administrator (sa) account.

Enter password:

Confirm password:

Specify SQL Server administrators

WIN-MC5FVTPMRR\vagrant (vagrant)

SQL Server administrators have unrestricted access to the Database Engine.

Add Current User Add... Remove

Рисунок 10. Настройка аутентификации

На вкладке **Data Directories** можно поменять при необходимости местоположение файлов баз данных по умолчанию:

Server Configuration | **Data Directories** | TempDB | FILESTREAM

Data root directory: C:\Program Files\Microsoft SQL Server\ ...

System database directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Data

User database directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Data ...

User database log directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Data ...

Backup directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Backup ...

Рисунок 11. Настройка местоположения файлов

На вкладке **TempDB** можно изменить параметры базы данных **tempdb** по умолчанию:

Server Configuration | Data Directories | **TempDB** | FILESTREAM

TempDB data files: tempdb.mdf, tempdb_mssql_#.ndf

Number of files: 2

Initial size (MB): 8 Total initial size (MB): 16

Autogrowth (MB): 64 Total autogrowth (MB): 128

Data directories: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Data Add... Remove

TempDB log file: templog.ldf

Initial size (MB): 8 Setup could take longer with large initial size.

Autogrowth (MB): 64

Log directory: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Data ...

Рисунок 12. Настройка tempdb

12. Analysis Services

Выберите **"Multidimensional and Data Mining Mode"** и нажмите кнопку **[Add Current User]**.

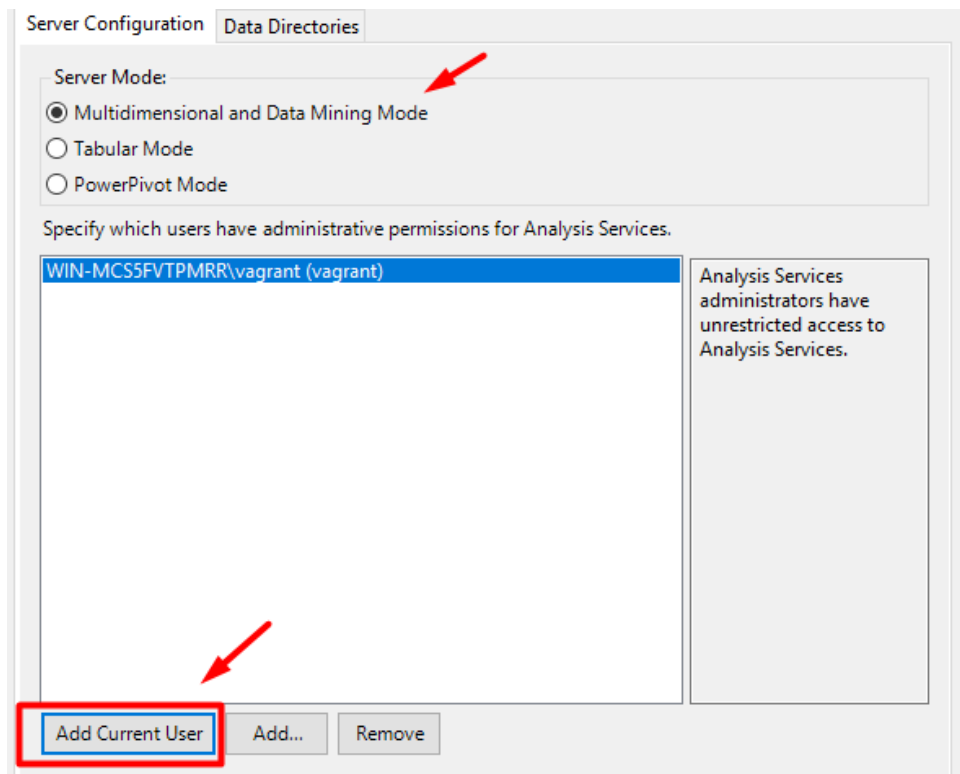


Рисунок 13. Конфигурация Analysis Services

13. Нажмите **[Install]**.

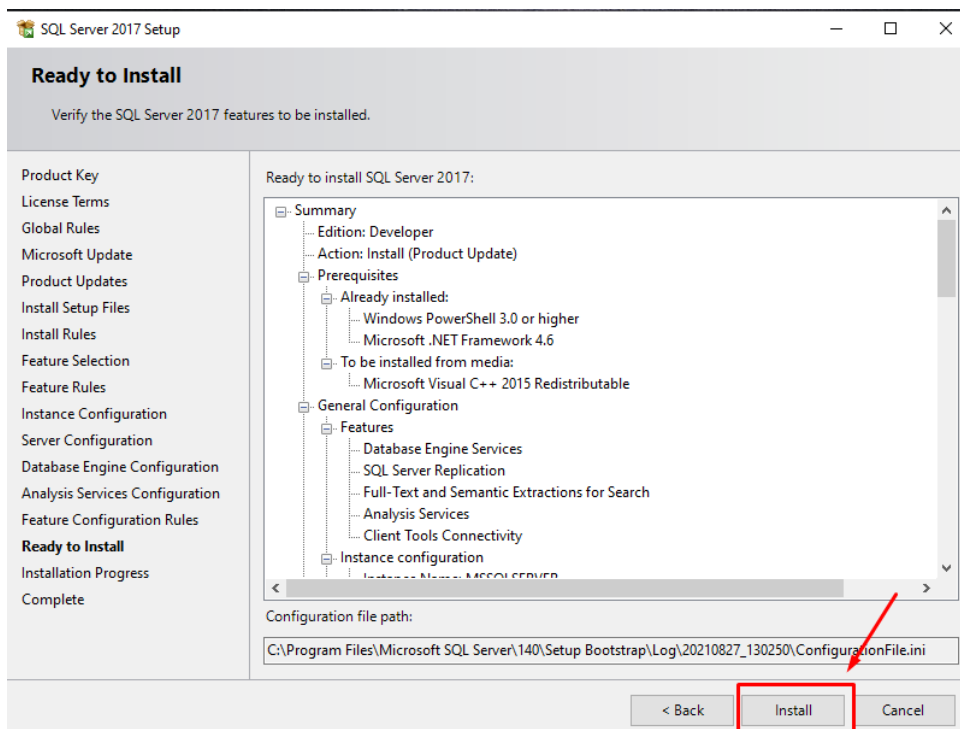


Рисунок 14. Установка

2. SQL Server Configuration Manager

В последних версиях [SQL Server Configuration Manager](#) реализован как MMC-консоль.

Для SQL Server 2017 можно запустить следующим образом:

```
C:\Windows\SysWOW64\SQLServerManager14.msc
```

Для других версий SQL Server:

- SQL Server 2019 — `C:\Windows\SysWOW64\SQLServerManager15.msc`
- SQL Server 2017 — `C:\Windows\SysWOW64\SQLServerManager14.msc`
- SQL Server 2016 — `C:\Windows\SysWOW64\SQLServerManager13.msc`
- SQL Server 2014 — `C:\Windows\SysWOW64\SQLServerManager12.msc`
- SQL Server 2012 — `C:\Windows\SysWOW64\SQLServerManager11.msc`

Или добавить в саму MMC-консоль.

Или найти в "Управление компьютером".

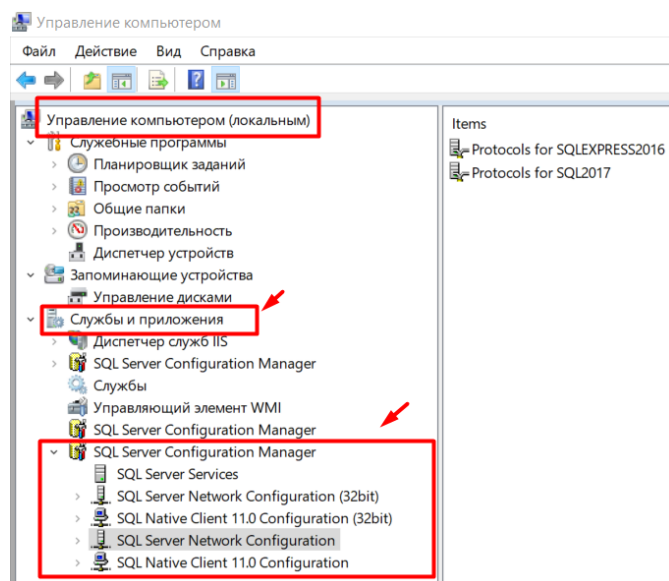


Рисунок 15. SQL Server Configuration Manager в Управление компьютером

3. SQL Server Management Studio (SSMS)

SQL Server Management Studio (SSMS) не входит в комплект дистрибутива SQL Server 2017. Скачивается отдельный инсталлятор.



Вместе с SSMS устанавливается **Azure Data Studio**.

<https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms>

Также ссылку можно найти в **SQL Server Installation Center (Installation > Install SQL Server Management Tools.)**.

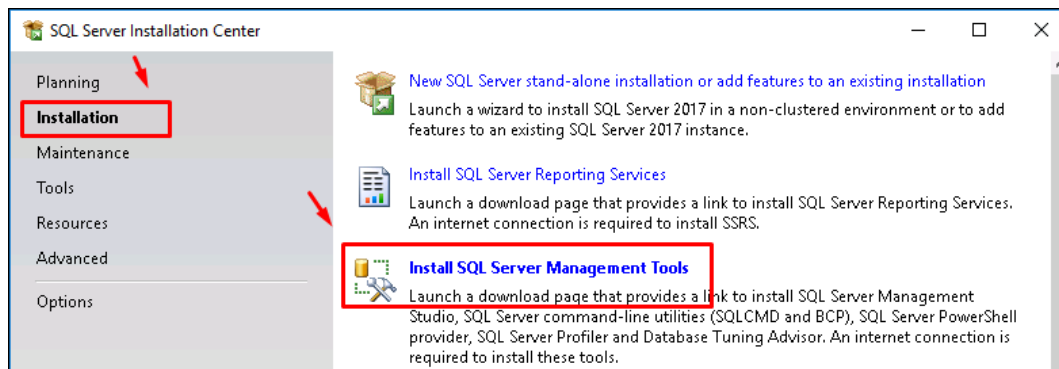


Рисунок 16. SQL Server Installation Center

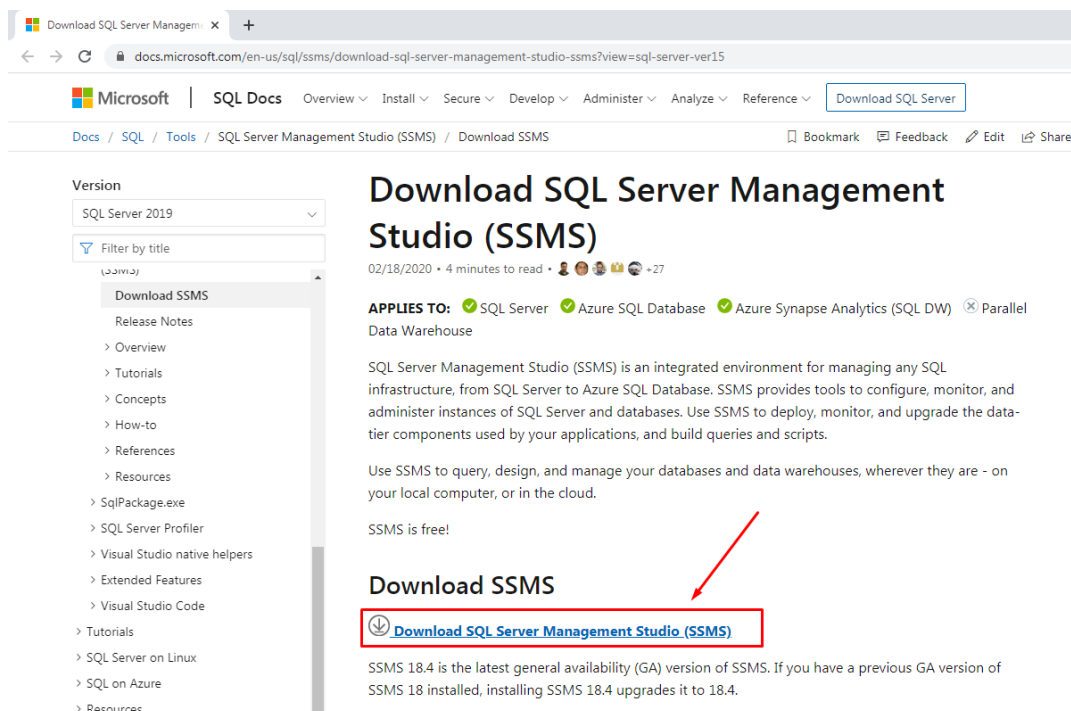
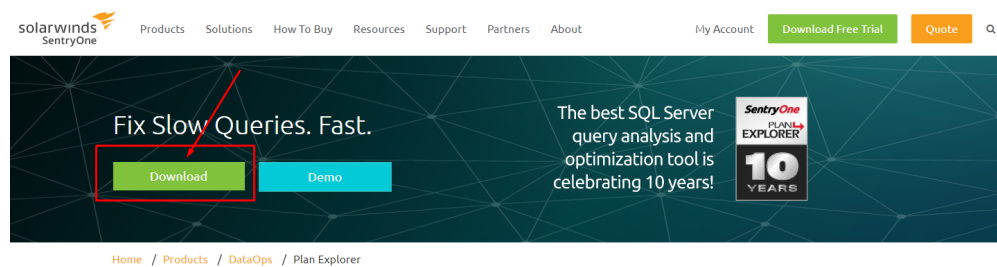


Рисунок 17. Страница скачивания SSMS

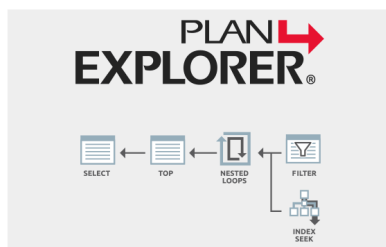
4. SQL Sentry Plan Explorer

SQL Sentry Plan Explorer используется для анализа плана выполнения запросов.

<https://www.sentryone.com/plan-explorer>



Get Advanced Query Tuning Features Not Found in Any Other Tool



- Index Analysis
- Statistics Analysis with Histogram
- Query Performance Profiling and Playback
- Actual Plan Recosting
- Superior Plan Diagram
- SSMS Add-In
- Tuning History and Comments
- Deadlocks Visualization
- And more
- Always Free

Рисунок 18. SQL Sentry Plan Explorer

Ставится простым [Next], [Next], ...

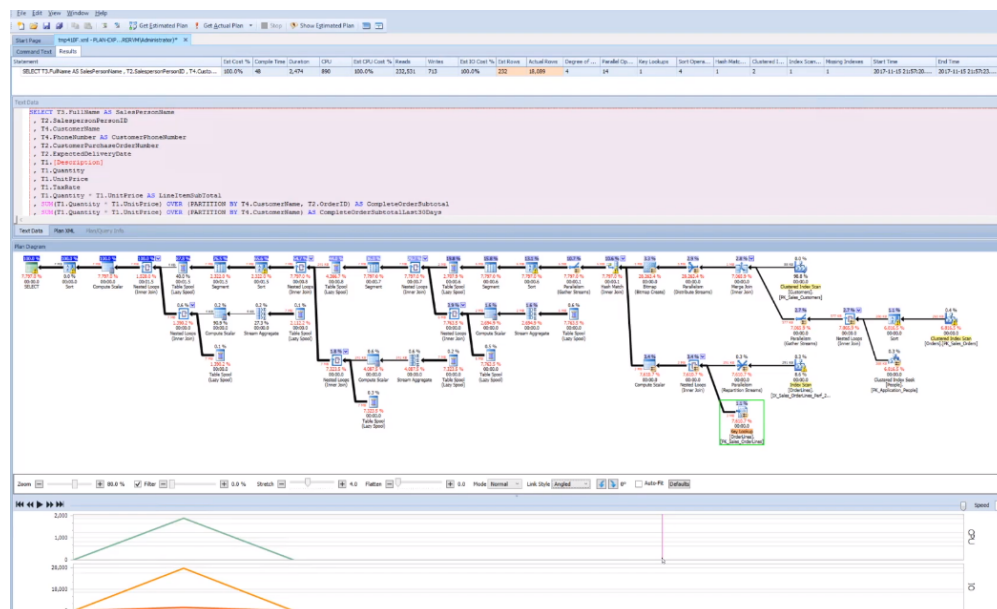


Рисунок 19. Пример плана выполнения в SQL Sentry Plan Explorer

5. Visual Studio 2019 Community

Visual Studio понадобится для работы с CLR (разработка хранимых процедур и др на C#), для проектов Analysis Services, SQL Server Integration Services и SQL Server Reporting Service.



Visual Studio сейчас устанавливать не обязательно. Можно установить позже, когда будет соответствующее занятие (второй модуль).

1. Ссылка <https://visualstudio.microsoft.com/vs/community/>
2. Выбрать **Data storage and processing** (Хранение и обработка данных).

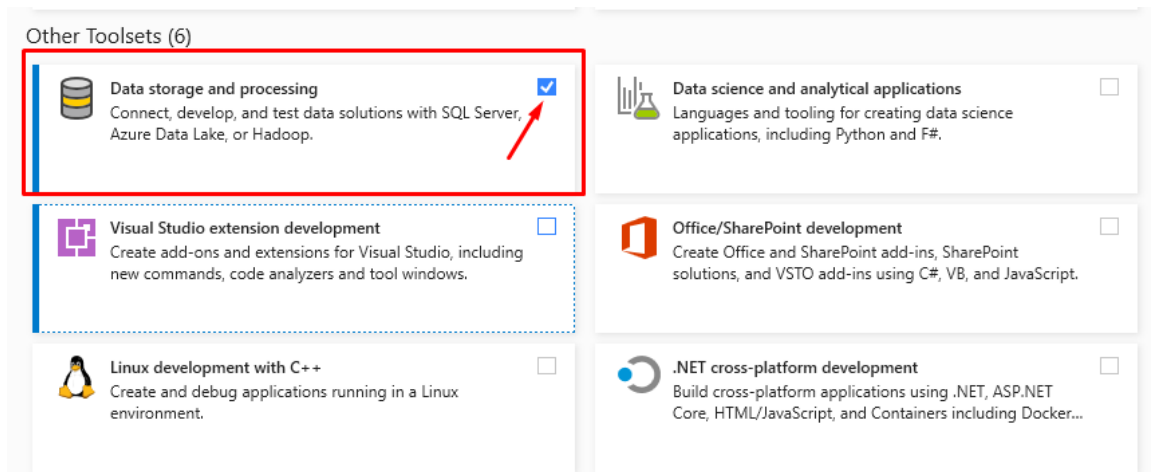


Рисунок 20. Установка Visual Studio — Data storage and processing

3. После установки Visual Studio установите в нем расширение для проектов SQL Server Analysis Services (SSAS): **Extensions** > **Manage Extensions**.



Рисунок 21. Меню Manage Extensions

В поиск введите "ssas", найдите и установите расширение **Microsoft Analysis Services Projects**.

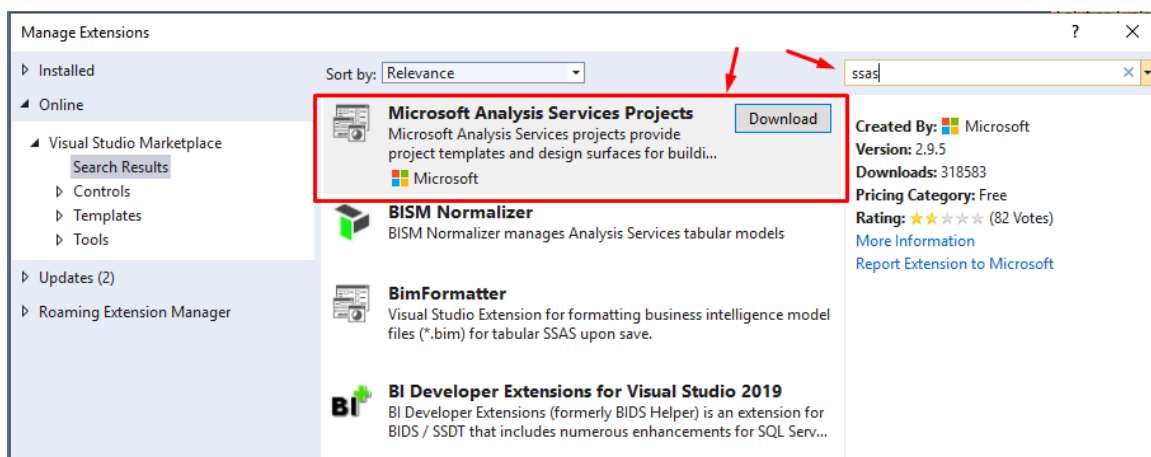


Рисунок 22. Установка расширения Microsoft Analysis Services Projects

4. Должны появиться проекты Analysis Services.

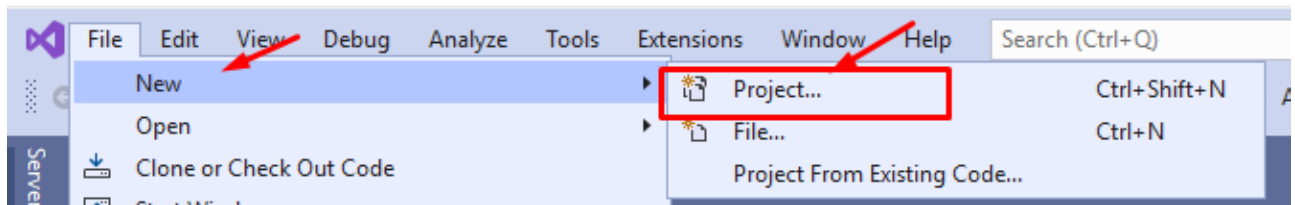


Рисунок 23. File → New → Project...

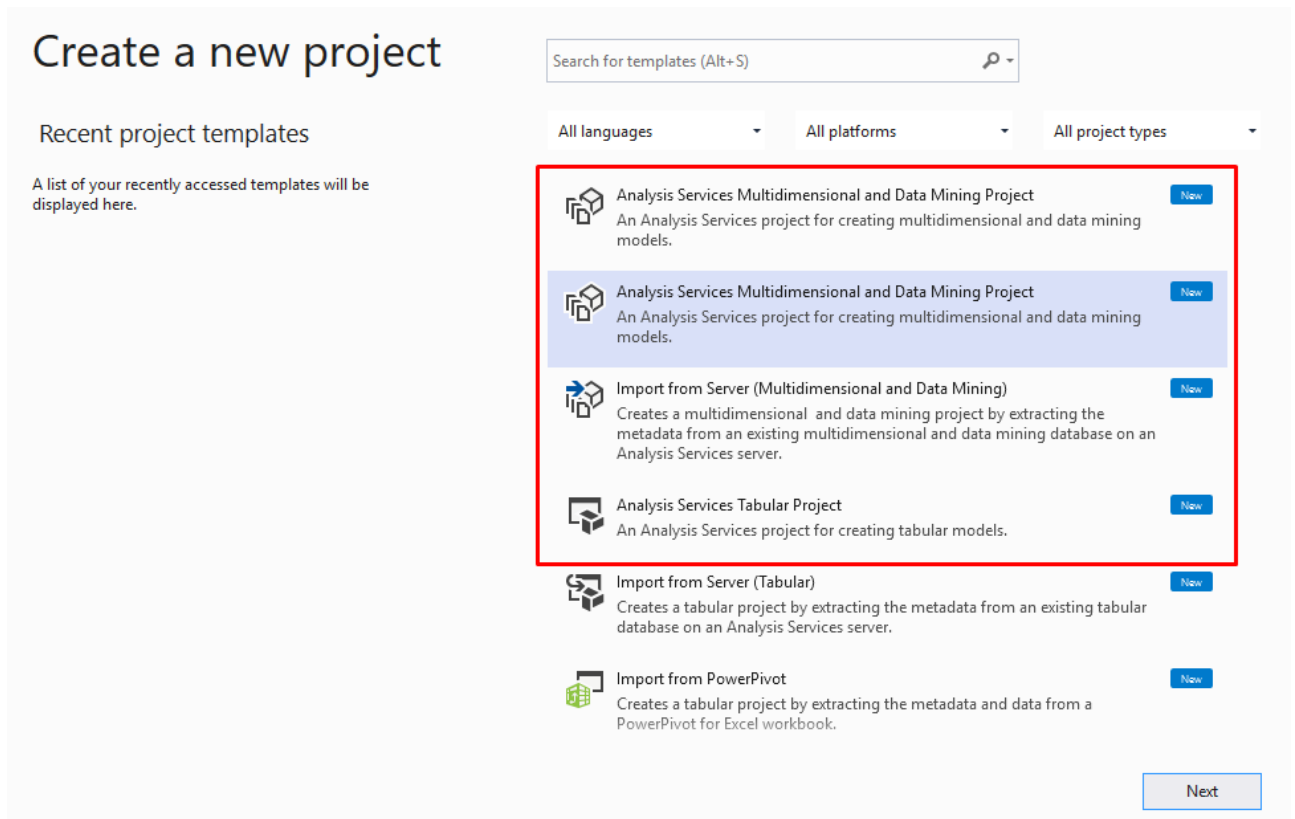


Рисунок 24. Проекты Analysis Services

6. БД WideWorldImporters

Для демонстрации и домашних заданий используется демонстрационная база данных **WideWorldImporters** от Microsoft.

6.1. Разворачивание БД WideWorldImporters

1. Скачайте бэкап БД — файл **WideWorldImporters-Full.bak**

<https://github.com/Microsoft/sql-server-samples/releases/tag/wide-world-importers-v1.0>



Не перепутайте файлы. Нужен именно **WideWorldImporters-Full.bak**.

Файл "DW", **WideWorldImportersDW-Full.bak** — это другое, это не нужно.

Wide World Importers sample database v1.0



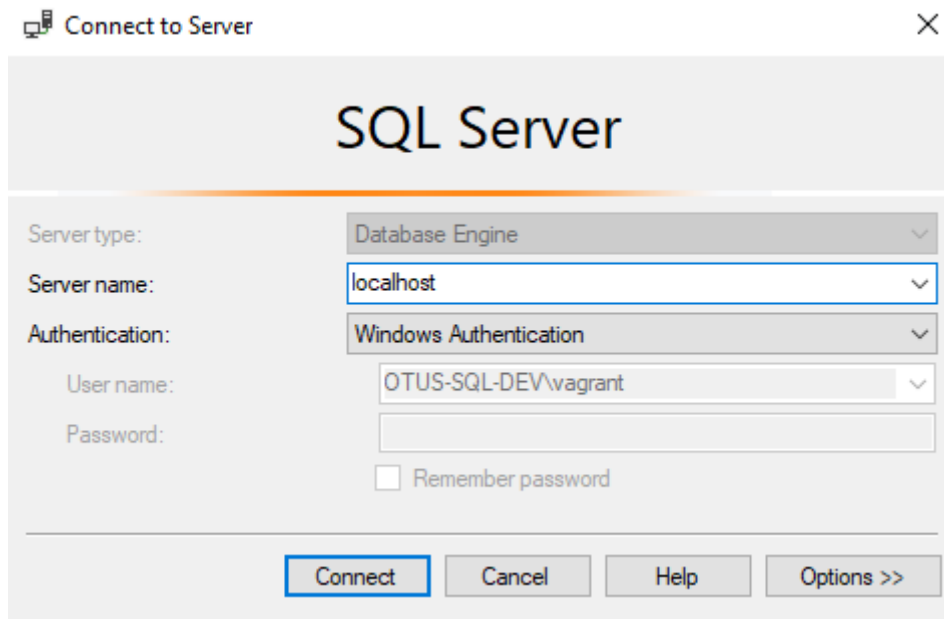
jodebrui released this on Jun 8, 2016 · [1818 commits](#) to master since this release

WideWorldImporters sample database for SQL Server (starting 2016) and Azure SQL Database. For details see the corresponding [documentation](#).

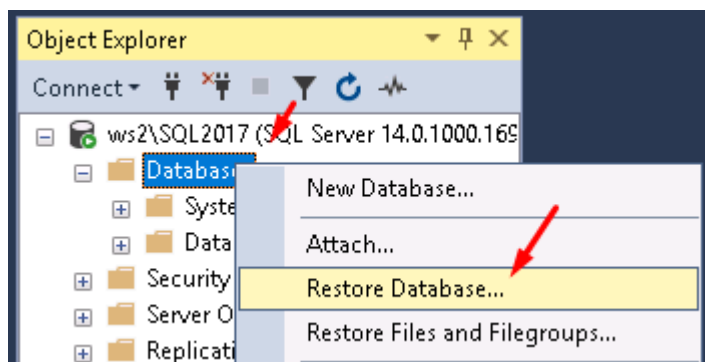
This release includes the following artifacts:

- **SQL Server 2016 SP1 (or later) Any Edition aside from LocalDB; SQL Server 2016 RTM (or later) Evaluation/Developer/Enterprise Edition**
 - **WideWorldImporters-Full.bak** - full sample database for OLTP (OnLine Transaction Processing) and HTAP (hybrid transaction and analytical processing aka real-time operational analytics). For SQL Server 2016 SP1 (or higher), any edition.
 - Update (8/12/2016): updated stats on system tables to support efficient reverse engineering with Entity Framework
 - **WideWorldImportersDW-Full.bak** - full sample database for OLAP (OnLine Analytical Processing). For SQL Server 2016 SP1 (or higher), any edition.
 - Hardware requirements: installing both databases requires 10GB of available disk space, and 1.5GB of available memory for SQL Server.

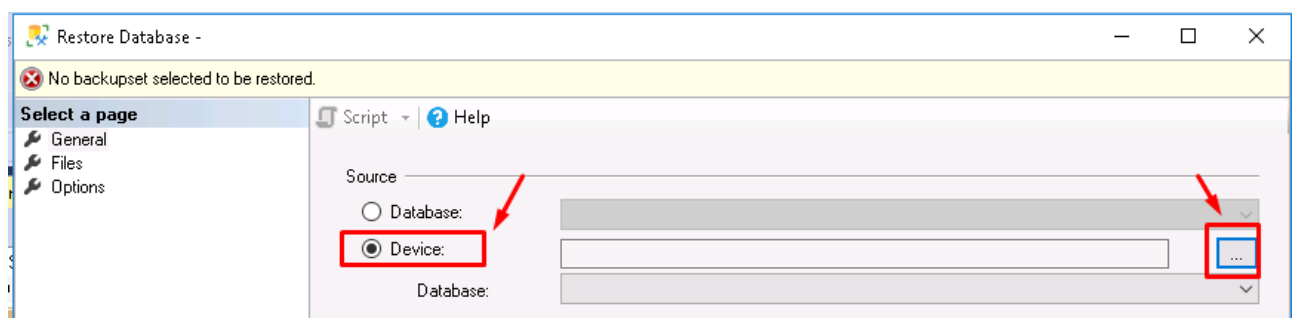
2. В SQL Server Management Studio (SSMS) подключитесь к установленному SQL Server.



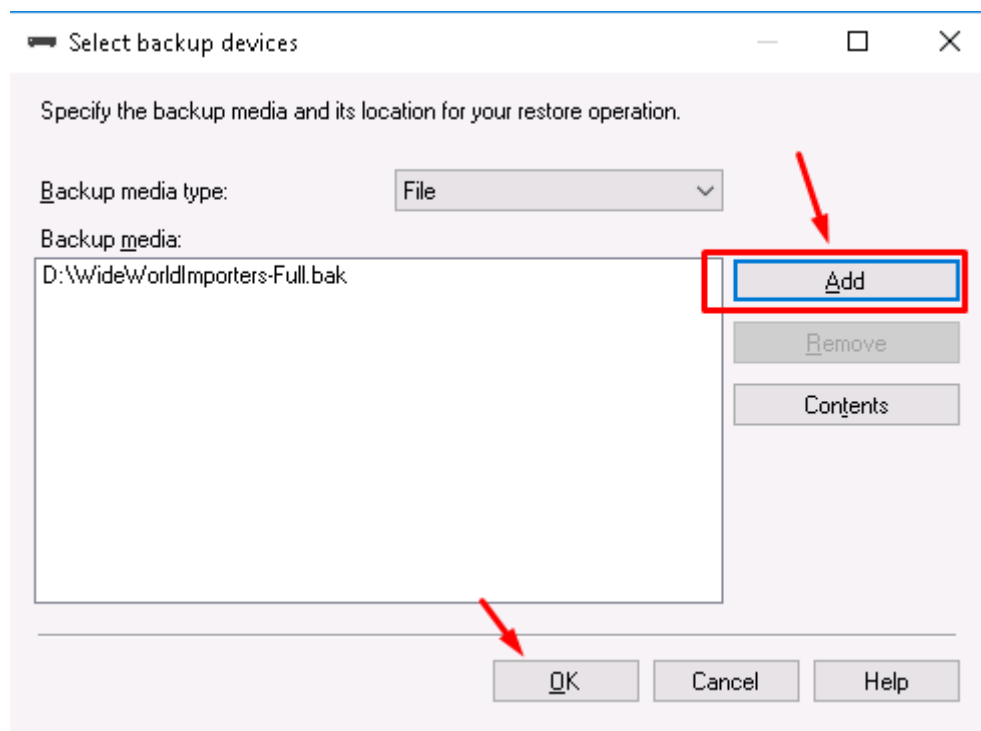
3. В SSMS выберите **Databases** > **Restore Database**.



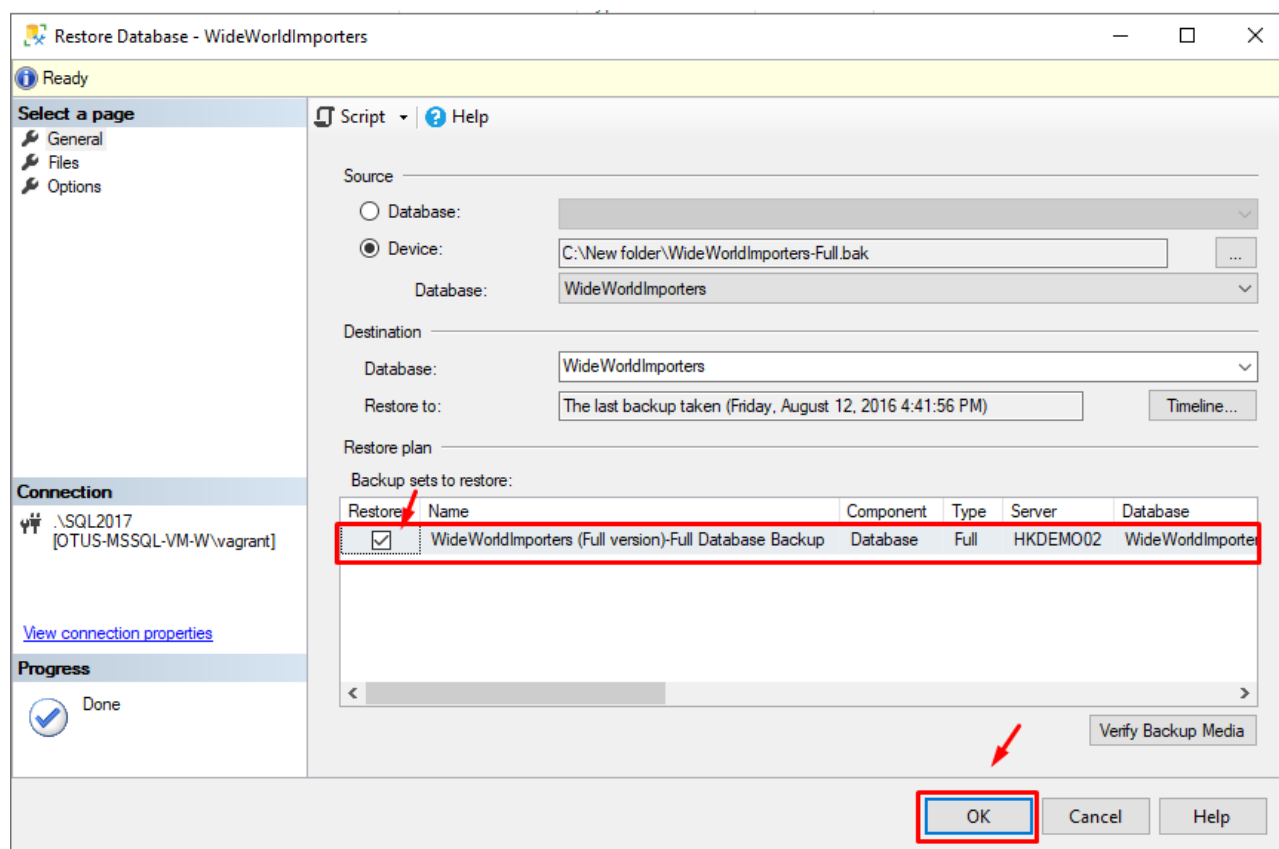
4. В окне "Restore Database" выберите **Device** и нажмите кнопку .



5. Выберите скачанный файл **WideWorldImporters-Full.bak**.



6. В списке "Backup sets to restore" должен появиться бэкап. Выберите его и нажмите [OK].





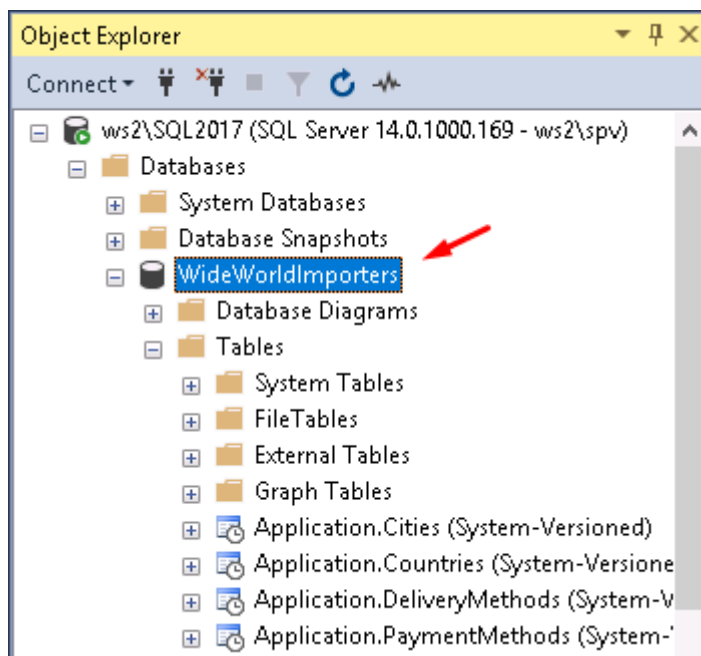
Если будет ошибка **Operating system error 5(Отказано в доступе) RESTORE HEADERONLY is terminating abnormally**

'Cannot open backup device 'C:\Program Files\Microsoft SQL Server\MSSQL14.SQL2017\MSSQL\Backup\WideWorldImporters-Full.bak'. Operating system error 5(Отказано в доступе.).RESTORE HEADERONLY is terminating abnormally. (Microsoft SQL Server, ошибка: 3201)'

То пользователю, под которым запускается SQL Server, надо дать права на папку с бэкапом или перенести файл бэкапа (**WideWorldImporters-Full.bak**) в то место, где есть права.

<https://blog.sqlauthority.com/2011/04/13/sql-server-fix-error-msg-3201-level-16-cannot-open-backup-device-operating-system-error-5access-is-denied/>

7. В списке Databases появится БД **WideWorldImporters** (возможно понадобится нажать [F5] или [Refresh] в контекстном меню **Databases**).



6.2. Описание БД WideWorldImporters

Описание БД **WideWorldImporters** от Microsoft:

- Wide World Importers sample databases for Microsoft SQL
<https://docs.microsoft.com/en-us/sql/samples/wide-world-importers-what-is>
- WideWorldImporters database catalog
<https://docs.microsoft.com/en-us/sql/samples/wide-world-importers-oltp-database-catalog>

7. Git / GitHub

GitHub используется для сохранения домашних заданий.

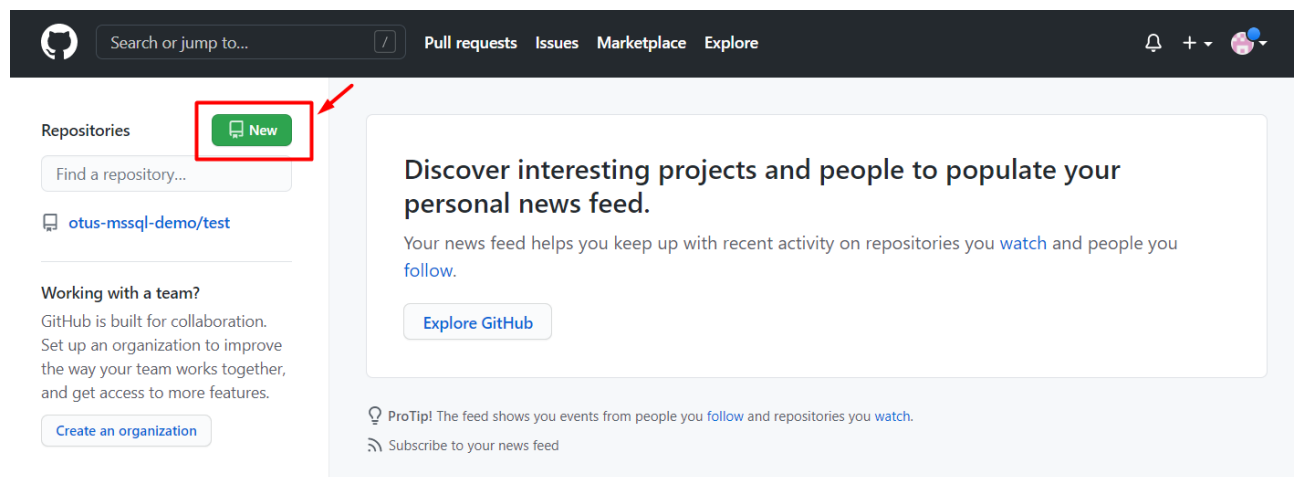
После выполнения ДЗ и сохранения его в github в "Чат с преподавателем" отправляете ссылку на ваш репозиторий.

7.1. Установка git в Windows

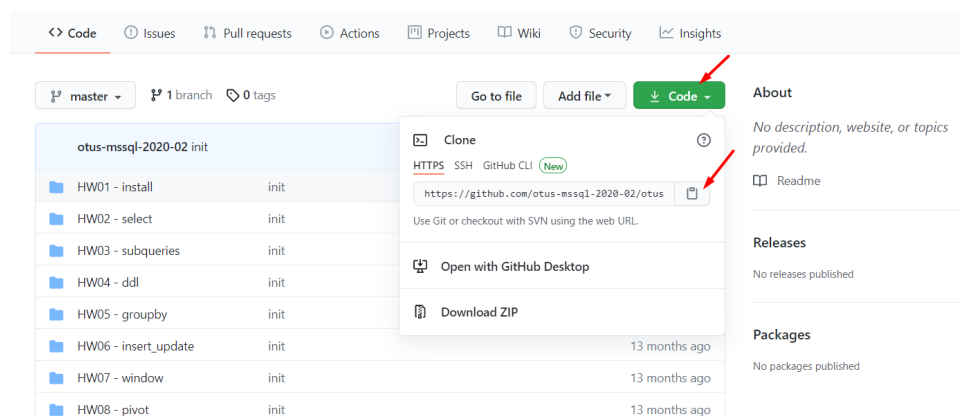
1. Скачать, установить git — <https://git-scm.com>
2. Скачать, установить TortoiseGit — <https://tortoisegit.org>

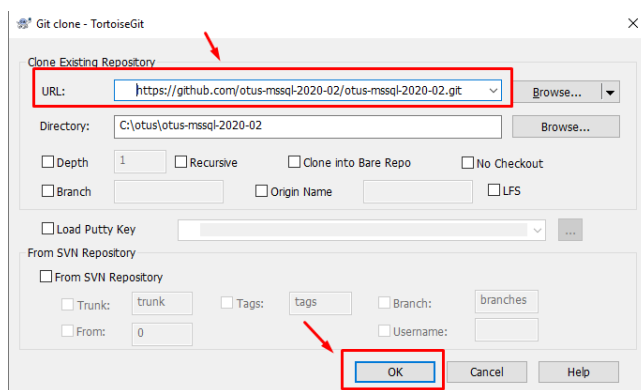
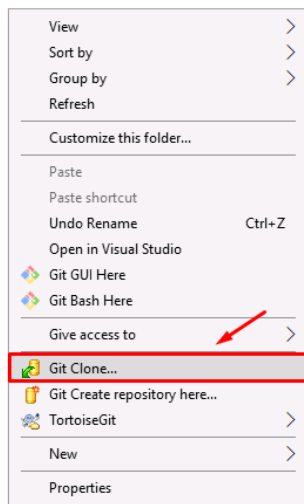
TortoiseGit — это удобный GUI для git, встраивается в контекстное меню проводника Windows. TortoiseGit опциональный, можете работать через командную строку или через IDE или через любой другой GUI, но ниже примеры с TortoiseGit.

3. Зарегистрируйтесь на [github](https://github.com) (можете использовать существующую учетную запись — на ваше усмотрение).
4. Создайте новый репозиторий `otus-mssql-<ваше_имя>` (например, `otus-mssql-ivanov`).



5. Клонировать репозиторий себе на компьютер:





Добавьте **README.md** с содержанием:

Домашние задания курса OTUS "MS SQL Server Developer".
Группа 2021-12.

Добавьте файл **.gitattributes** (без этого git сравнивает sql-файлы как бинарные):

```
*.sql diff
```

6. Работаете, редактируете, создаете папки как с обычными файлами.
7. Для заливки на github надо сделать сначала **commit**, потом **push**.

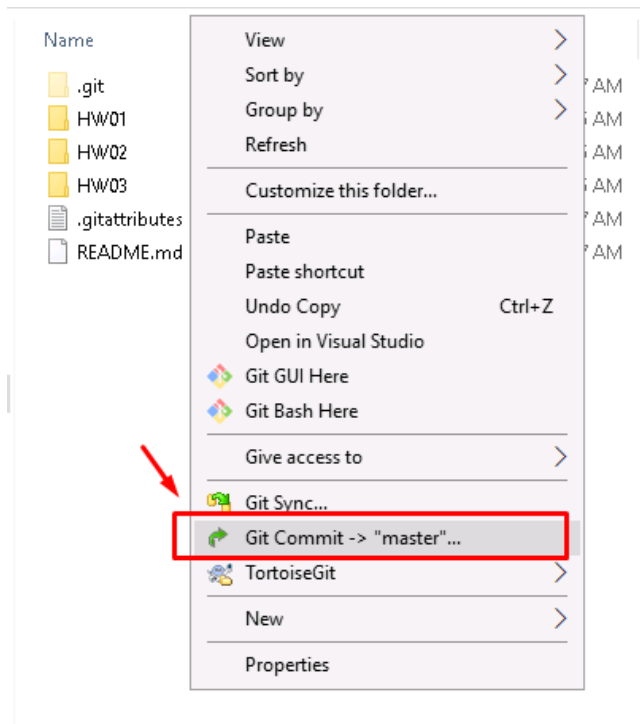


Рисунок 25. Tortoise Git — Commit

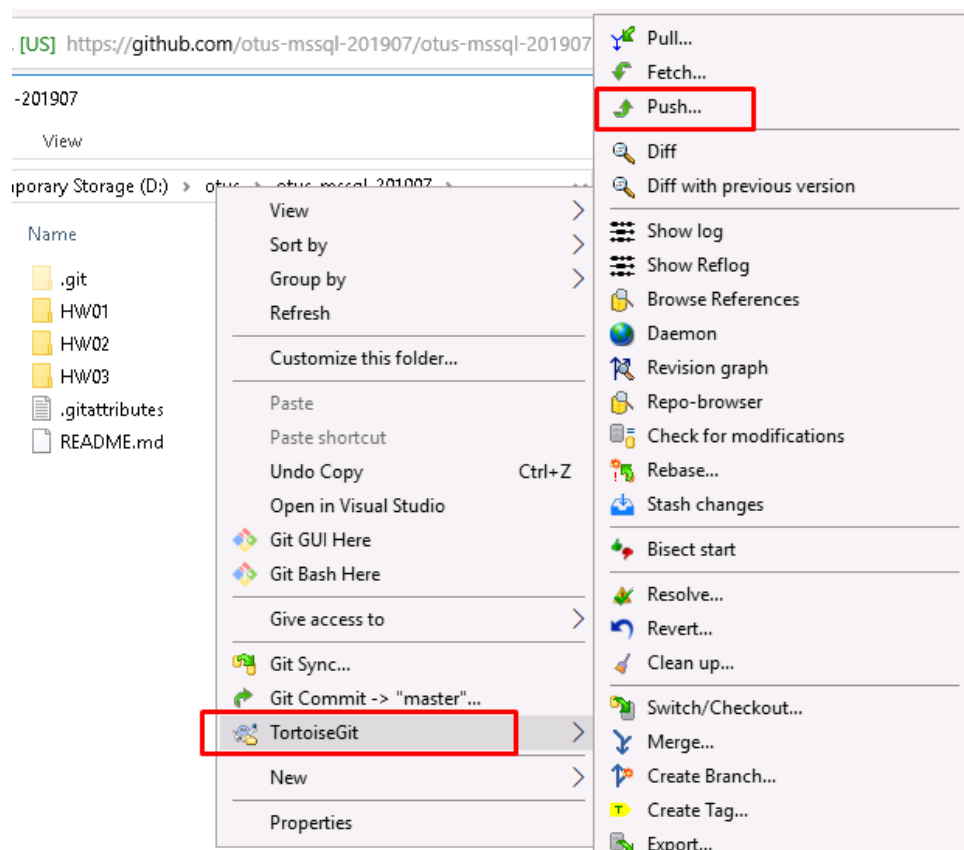


Рисунок 26. Tortoise Git — Push

Или через консоль:

- Клонировать репозиторий:

```
git clone <ссылка на ваш репозиторий>
```

- Вносите правки, работаете над проектом, делаете commit:

```
git commit -m "your comment" -a
```

- Отправляете в github через push:

```
git push
```

7.2. Рекомендации к оформлению репозитория

Для того чтобы удобнее проверять домашние работы и ориентироваться внутри репозитория, в т.ч. потенциальным работодателям, желательно, чтобы он удовлетворял следующим требованиям:

1. Название репозитория должно быть в формате **otus-mssql-XXXX**, где
 - **XXXX** — фамилия/ник слушателя (например, **ivanov**).
2. Для проверки будет лучше, если каждая домашняя работа будет сдаваться в новой папке внутри репозитория (например, **HW02 - select** и тп).
3. Желательно, чтобы sql-скрипт с выполненным заданием содержал в комментариях текст задания.
4. Желательно сохранять файлы в UTF.

Пример структуры репозитория можно посмотреть здесь <https://github.com/otus-mssql-2020-02/otus-mssql-2020-02>

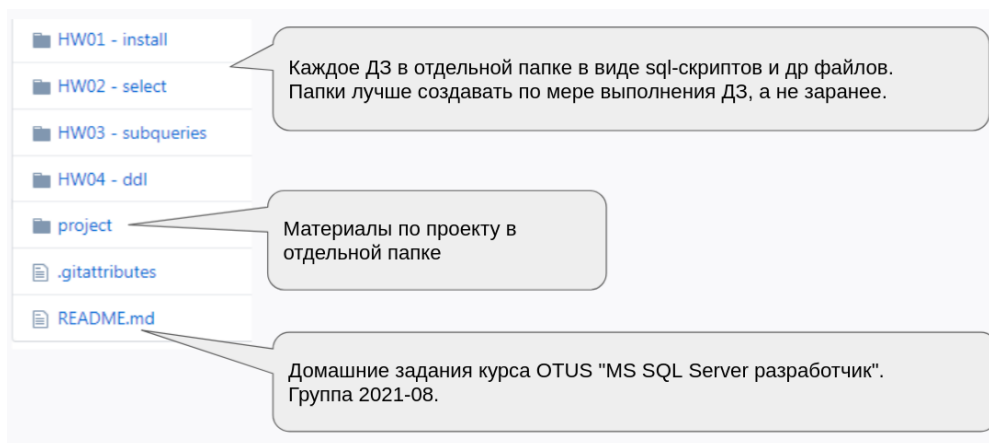


Рисунок 27. Пример структуры репозитория

7.3. Ресурсы для изучения git / github

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