

Your fastest way to learn. Why wait?



Microsoft

MCSA: SQL Server

Pre-course Reading

---

R-481-01

Version 1.0

<https://firebrand.training>

# Pre-course Reading

So you are taking a course about SQL from Firebrand. As you may already know there are three different tracks for studying SQL 2016 offered at Firebrand:

- MCSASQLDD - a track for database developers
- MCSASQLDA - a track for database administrators
- MCSASQLBID - a track for business intelligence developers

The track for Database Development will entail writing code in Transact SQL ranging in complexity from a simple Select statement to retrieve the data stored in a table or tables to creating more complex programming objects like Stored Procedures, Triggers, Functions and the like.

The Database Administration track covers items like how to create logins and users, do backup and restore as well as more complex items like index management and creating an Azure database out in the cloud.

The Business Intelligence track includes developing ways to move data into a data warehouse and then using that data warehouse as a data source for models that your business users can run reports against.

In this document there will be sections for each of the tracks including some links to some helpful websites that can help you prepare for your upcoming Firebrand course.

Before we break down into the different subsections for the different tracks, let's start off with a little history about Microsoft SQL Server.

Many years ago Microsoft bought a product that was called Sybase and it became Microsoft SQL Server. It was a multiuser relational database which means it provided a way to hold data in tables that were related to each other and allowed multiple users to access that data simultaneously. In a relational database tables are joined together by relationships known as foreign keys. Each table can also have a field which uniquely identifies each row in the table called a primary key. In a relational database the tables can have indexes created for them to make querying them more efficient. There are generally two types of indexes, ones that contain all the data in the entire table known as a clustered index and those that contain only certain columns from a table known as non-clustered indexes.

There are many vendors that offer a relational database product, Oracle, Microsoft and others.

The courses that will be described below are focusing on Microsoft's version of a multi-user relational database. Microsoft's version of a relational database has undergone many different versions in the years since they purchased the product from Sybase and some are listed below.

1.0 (OS/2)	1989	SQL Server 1.0 (16-bit)	Filipi
4.2A (OS/2)	1992	SQL Server 4.2A (16-bit)	-
6.0	1995	SQL Server 6.0S	QL95
6.5	1996	SQL Server 6.5	Hydra
7.0	1998	SQL Server 7.0	Sphinx
8.0	2000	SQL Server 2000	Shiloh
9.0	2005	SQL Server 2005	Yukon
10.0	2008	SQL Server 2008	Katmai
10.5	2010	SQL Server 2008 R2	Kilimanjaro (aka KJ)
11.0	2012	SQL Server 2012	Denali
12.0	2014	SQL Server 2014	SQL14
13.0	2016	SQL Server 2016	-

There is a language to get data out of the tables in a relational database and that language is called Structured Query Language or SQL. And just as there are different versions of a relational database there are different versions of SQL and Microsoft's version is called T-SQL or Transact SQL.

The course that focuses on Database Development will be about writing T-SQL code, not only to query or retrieve data from its storage location (tables) but also to create objects to manage inserting new data and other tasks.

If data is being stored in tables, someone has to manage who can access that data, how much of that data can a person access and what can a person do to that data, just read it or change it. In addition to controlling permissions and data access, a database administrator must make sure that queries run as efficiently as possible and that the data stays available. So part of what a database administrator has to do is deal with concepts like high availability or failover clustering, backups and restores, creating indexes to make queries run faster and keeping those indexes in good working order, also known as index defragmentation. So while some of these jobs can be done through user interfaces known as GUIs, there will still be a need to understand and use code to do the job of a database administrator. The code used could be T-SQL code or it could be another code used by Microsoft across a great many of the products that they offer which is called PowerShell.

In creating relational databases there are a series of standards that control how a table is created, what kind of columns it can hold, and those rules are called normal forms. In the track that deals with Database Development, the tables that are created will be those that make transactions (inserting, updating and deleting data) faster and create less locking of data. However, if one is designing a data warehouse, the table structure often violates those rules or normal forms to make querying that data more efficient. That process is called denormalization.

[Link with explanations about normal forms](#)

Many years ago a way of looking at data that was collected over time by a business was designed and it is called business intelligence. In this field there have been some major creators or designers, people like Ralph Kimball and Bill Inmon. Each of these people have expressed ways that a data warehouse should be designed or

could be designed. So in the track dealing with Business Intelligence what should be the table structures in a data warehouse will be covered, also how should the tables be related to each other (star schema or snowflake schema), what kind of data should go in a fact table vs. a dimension table, what kind of relationships can exist in a data warehouse and many more topics. In addition to how to design the data warehouse, the Business Intelligence track will cover how to create an additional item that business users can query to look at the data that doesn't lock records in the data warehouse. Those items are called models. Those models are created by a tool called Analysis Services.

In 1998, with SQL version 7.0, Microsoft released a product known as Analysis Services or OLAP which stands for Online Analytical Processing. That tool was used to create a model or separate storage location for the data stored in a data warehouse and a new way of looking at that data called the Multi-dimensional model or cube. With a cube, a person could create hierarchies to be able to organize how to look at the data, Key Performance Indicators to look at specific value and trends or do data mining to look at past behaviour or who best to market to.

Although Microsoft and others provide a number of different ways to get data out of a cube, like Pivot tables or Charts in Excel or in Reporting Services, the language to query a cube manually (MDX - Multidimensional Expressions) is fairly complex and has a large learning curve, especially for business users. So in SQL 2012 Microsoft released a new type of Analysis Services called the Tabular Model. The tabular model is considered by most to be easier to configure and the language to manually query the tabular model (DAX - Data Analysis Expressions) is much more like formulas in Excel a product that most business users have some experience with.

Microsoft offers a product to move data into a data warehouse from a database where transactions take place (OLTP - Online Transactional Processing) and that tool is called SQL Server Integration Services. Prior to SQL 2005, the product that an administrator could use to load data from a transactional database to a data warehouse was called DTS (Data Transformation Services). But since SQL 2005 the tool to load data from one location to another is SSIS or SQL Server Integration Services. This tool can not only move data; it can change the format of the data (transform) or redirect the data to different locations based on the value of the data (conditional split) as well as a host of other tasks. The tasks that need to be performed on the data are encapsulated in a package. As part of the Business Intelligence track, we will use SSIS to create many packages going over many of the features and functions that can be performed using Integration Services.

There are two additional tools that can help an administrator in data movement or in dealing with the quality of the data that is being received and those tools are Data Quality Services and Master Data Services. Data Quality Services can be used to both clean data that might have been typed incorrectly and also to de-duplicate data that is coming from multiple sources that might have the same record in multiple places. Master Data Services can be used to create a central staging or

storage location for data coming from multiple transactional sources. Both tools will be covered in the Business Intelligence track.

Microsoft also offers a tool not covered in any of the above mentioned tracks that can be used to build and deploy reports; that tool is called Reporting Services. It can either be installed as a stand-alone product, known as Native Mode or as part of a SharePoint portal, known as SharePoint Integrated Mode. With a portion of Visual Studios known as SQL Server Data Tools, one can build projects either using Integration Services to move data, Analysis Services to build models or Reporting Services to build reports that can then be seen either as part of a SharePoint portal or in Report Manager (the web site created for Reporting Services in Native Mode).

As you can see, Microsoft SQL is a product with many components and every time Microsoft releases a new version, they may make changes to any portion of the product, the Database Engine that holds the data in tables and hold programming objects like stored procedures and is used to administer or manage the databases, Analysis Services whether in Multidimensional Mode or Tabular Mode, Integration Services, etc. Below are links that you can follow to see what has changed in the different components with the release of SQL 2016.

[New Features of the Database Engine SQL 2016](#)

[New Features in Analysis Services SQL 2016](#)

[New Features in Integration Services SQL 2016](#)

[New Features in Master Data Services SQL 2016](#)

[New Features in Reporting Services SQL 2016](#)

In the link for what are the new features for Analysis Services in 2016, Microsoft has added a link to an online virtual lab environment where you can practice using some of the new features of Analysis Services for the Tabular Model. That link will also give you a look at SQL Management Studios, the primary tool that is used to create databases and query databases and manually run either MDX queries or DAX queries. It will also allow you to practice creating a Tabular Model.

## **Additional Reading for Database Developers:**

[T-SQL Tutorials](#)

The link starts at lesson 1 which is a basic T-SQL Select statement and then progress through lesson 10 that goes over how to create a stored procedure. With this link you can either just read the code discussed or download the code and try it on your own SQL server if you have one.

### [A More Advanced Tutorial](#)

This link has some more advanced T-SQL code that you can look at the syntax of, to see some of what will be covered in the Database Developer track.

## Additional Reading for Database Administrators:

### [Free E-books on Database Administration](#)

This link gives you access to eight free e-books that cover topics like Memory Optimized tables, how to integrate Big Data into your SQL environment, how to read execution plans which can be helpful for both database administrators and database developers, how to tune your SQL server's performance and many other topics. They do cover versions of SQL other than SQL 2016 but topics that are applicable also to SQL 2016.

### [Free E-book from Microsoft on SQL 2016](#)

This site is a location at Microsoft to get a book on the features of SQL 2016 in a pdf format. At the site Microsoft will ask for your name and location before letting you download the pdf document.

### [Beginner's Guide to SQL Azure](#)

This site gives some introductory information about Microsoft's online database product known as SQL Azure. The primary focus of the second test in the Database Administrators track is SQL Azure.

## Additional Reading for Business Intelligence Developers:

### [SSIS Tutorial](#)

This site contains a couple of lessons on introducing SSIS and also showing you how to create a basic SSIS package using the Import Export Wizard and SSDT (SQL Server Data Tools)

### [Analysis Services Tutorials](#)

This Microsoft site contains many tutorials on how to use Analysis Services both Multidimensional Mode and Tabular mode.

### [Simple MDX Examples](#)

This site shows a number of MDX examples for someone starting to use MDX.

### [DAX Query Examples](#)

This site shows some DAX query examples to start learning how to use DAX to query a Tabular Model.

## Glossary of Terms:

Heap:	The storage mechanism for an unsorted table without a clustered index
Clustered Index:	The storage mechanism for a sorted table
Non-clustered Index:	An index that doesn't hold all the columns of a table usually created to aid in joining table together in a query
Query:	An SQL statement to return data to the end user
Row-store Index:	An index that stores and sorts the data as rows
Columnstore Index:	An index that stores data in columns
Primary Key:	A value that uniquely identifies each row in a table, can be just on a single column or a combination of columns (composite primary key)
Foreign Key:	A relationship between tables that helps with joining tables together in a query and controls what values can be placed in that column. For example must be a valid Customer ID in the Customers table to be allowed into the Customer ID field in the Orders table
Normalized Database:	A database where the tables are designed with less columns, that follows the normal forms and make inserting or changing the data more efficient - transactional database
Denormalized Database:	A database where the tables are designed with more columns, that don't follow the normal forms but make querying the data more efficient - analytical database
Fact Tables:	Tables in the data warehouse that store primarily keys (whether primary or foreign) and numbers (called measures)
Dimension Tables:	Tables in a data warehouse that hold ways to look at the numbers (attributes) in the fact tables.
Star Schema:	A design for a data warehouse where all the dimension tables have a direct foreign key relationship with a central fact table
Snowflake Schema:	A design for a data warehouse where there are dimension tables that do not have a direct relationship to a central fact table but rather to another dimension table
MDX:	Multidimensional Expressions the query language used to retrieve data from the cells of a cube (tuple) or create objects in a cube
Cube:	The model type created by SQL Server Analysis Services in Multidimensional Mode
DMX:	Data Mining Extensions the query language used to retrieve data from a data mining model or create objects in that model, used to study trends and make predicts

ETL:	Extract Transform and Load, the process done by Integration Services when it is used to move data
High Availability:	Making sure the data is always available
Failover Clustering:	Setting up multiple servers to act like one server so if one node (one server in the cluster) goes down another one continues to do the work
Windows Failover Cluster:	Setting up a failover cluster at the operating system level
SQL Failover Cluster:	Setting up SQL to act like a failover cluster