作为一名程序员，对于SQL的使用算是基础中的基础，虽然也写了很多年的SQL，但常常还是记不清一些常见的命令，故而通过一篇博文巩固相关的记忆，并把T-SQL本身的一些新特性再进行一次学习。

首先回顾基础的概念，这部分可以跳过哈，比较枯燥。结构化查询语言SQL是基于集合理论和谓词逻辑的，大学课程中数字逻辑和离散数学主要会涉及这部分的内容。

集合理论是数学家Georg Cantor创建，是基于关系模型的数学分支。集合的定义为，任意集合体是我们感知或者想到的，能够确定的、**互异**对象m的整体。

谓词逻辑的渊源可以追溯到古希腊，也是基于关系模型的一个数学分支。创建关系模型领域的Edgar F. Codd博士提出通过谓词逻辑来管理和查询数据。一般来说，谓词是一个属性或是一个表示“持有”或者“不持有”的表达式，也就是“真”或者“假”。关系模型就是依靠谓词来维护数据的逻辑完成性并定义其结构的。在C#中，可以经常看到predicate，也是这个逻辑谓词。

关系模型涉及命题、谓词、关系、元组和属性等概念。在集合理论中，关系是集合的表现形式。在关系模型中，关系是相关的信息的集合，与SQL中相对应的就是表（而不是表间的关系）。需要注意的是，单个关系代表单个集合，而多个关系经过操作（基于关系代数）的结果会是一个关系，如联接操作。一般来说，在我们看来，谓词不是“真”，就是“假”。但让我们回忆一下，物理学中的一个著名实验“薛定谔之猫”，其主要思想就涉及“未可知态”，因此在数据库中，我们可以看到“true”，“false”，“unknown”三种情况。

同时，关系模型通过约束规则将定义数据完整性作为模型的一部分。常见的有提供**实体完整性**的候选键和提供**引用完整性**的外键。候选键是定义了一个和多个属性的键，防止关系中出现多个相同的元组，基于候选键的谓词可以唯一的标识行，可以在关系上定义多个候选键，通常会选择一个候选键作为主键，其他的候选键称为备用键。这儿可以补充的是，其实主键（候选键）的概念和索引的概念是分离的，只是数据库在实现时往往把主键作为聚集索引。外键用于强制引用完整性，外键定义了关系的一个或多个属性引用另一关系的候选键，此约束限定了引用关系的外键属性中的值，应该出现在被引用关系的候选键属性中的值。这儿想补充的是，在**高并发环境**下，一般不使用数据库外键，而是在业务层进行控制。此外还有非空约束、唯一约束等。

接下来介绍一个很常见但时间一长就容易忘记的概念，规范化规则（也叫范式）。规范化是一个常规的数学过程，用于确保每一个实体都由单一关系表示，以避免数据修改过程中的异常，并在不牺牲完整性的前提下保持最低的冗余。最常见的三个范式分别是：

1NF，表示关系中的元组必须是唯一的，而且属性是原子化的（这个往往根据需求而定，不绝对）。

2NF，在满足第一范式的基础上，对于每个候选键，每个非键属性必须是对整个候选键的完全函数依赖。简而言之，就是如果要获得任何非键属性值，需要提供相同元组候选键中所有属性的值，如果知道候选键的所有属性的值，就可以检索到任意元组的任何属性的任何值。

3NF，在满足第二范式的基础上，所有非键属性必须依赖于非传递的候选键。简单来说，就是所有非键属性间必须相互独立，一个非键属性不能依赖于另一个非键属性。

接下来简单介绍数据的生命周期，常见的业务系统一般仅仅包含联机事务处理一个阶段，但是随着业务规模的发展，会慢慢衍生出很多大数据方向的阶段，如图1所示，其中缩略词所代表的意思为：OLTP，联机事务处理；DSA，数据准备区；DW(OLAP)，数据仓库；BISM，商业智能语义模型；DM(Data Mining)，数据挖掘；ETL，提取、转换和加载；MDX，多维表达式；DAX，数据分析表达式；



图 1数据生命周期

其中，数据仓库主要是为了支持数据检索需求进行的数据模型设计和优化， 模型故意进行了冗余、减少表和简化关系。最简单DW结构是星型架构，包括多个维度表和一个事实表，每个维度表表示要分析的数据主题。例如在订单和销售系统中，可能要分析客户、产品、雇员、时间以及类似主题数据。**星型架构**中，每一个维度以冗余数据方式填充单个表（例如将Product、ProductSubCategory、ProductCategory合成一个ProductDim）。此外，如果想要规范化一个维度表，就会产生多个表来表示该维度，得到一个雪花维度，这种的结构也被称为**雪花架构**。

从源系统提取数据、处理数据并加载到数据仓库的过程，被称为提取、转换和加载ETL，SQL Server相关的产品就是我们常见SSIS，此过程常常涉及OLTP和DW之间数据准备区DSA的使用。

商业智能语义模型BISM提供丰富灵活的分析和报告功能，其体系结构包含三层，数据模型、业务逻辑和查询数据访问。模型可以部署在Analysis Services和PowerPivot上，前者针对BI专业人员，使用多维数据模型或表格，而后者针对企业用户，使用表格数据模型。业务和查询使用两种语言， 基于多维概念的多维表达式(MDX)和基于表格概念的数据分析表达式(DAX)。数据访问层可以从不同的数据来源获取数据：如DW这样的关系型数据库、文件、云服务、Odata订阅等。数据访问层既可以在本地缓存数据，也可以作为连接数据源的直通层。缓存模式下可以有两种存储引擎方式选择，一种被称为MOLAP的预先聚合方式，当初设计它是为了支持多维模型，而另一种称为VertiPaq的新引擎，它实现了列存储概念，具有很高的压缩级别和非常快速的处理引擎，不在需要预先聚合、索引等。

BISM为用户提供可能的答案，而DM为客户提供正确的答案。也就是说，数据挖掘算法梳理数据并筛选有用信息，Analysis Services支持的数据挖掘算法包括：聚类分析、决策树等。

关于SQL Server的一些例如ABC等产品概念就跳过，毕竟使用性不强，不过最近微软的云服务，包括数据库云服务（Window Azure SQL Database）还是有一些亮点的。技术概念比较多，之后一一介绍。

SQL Server实例的通过机器名\实例名唯一标示，例如Server1\Inst1。数据库实例中多个不同的数据库，系统数据库包括:master数据库存储实例范围的元数据信息、服务器配置等；model数据库用于创建数据的模板；tempdb数据库是存储临时数据的地方，如工作表、排序空间、行版本控制信息等；msdb数据库是一个称为“SQL Server代理”的服务存储数据的地方，SQL Server代理提供自动操作（包括作业、计划和警报）、复制服务、Database Mail、Service Broker、备份等。

数据库有数据文件和日志文件组成，在创建数据库时，可以为每个文件定义各种属性，包括文件名、位置、初始大小、最大大小和一个自动增长的增量。数据文件存储数据，日志文件维护事务信息。虽然SQL Server可以并行写入多个数据文件，但只能以连续的方式一次写入一个日志文件。因此，与数据文件不同，多个日志文件并不会带来性能的提升，同时如果日志所在磁盘空间不足，那么可能还需要添加日志文件，这点在实际工作中也经常遇到日志文件过大占满磁盘空间而造成数据库服务不可用。数据文件被组织在一个叫做“文件组”的逻辑组中，primary文件组包含XXX.mdf的主数据库文件，以及数据库系统目录，可以添加次要数据文件XXX.ndf到primary，同时XXX.ldf表示日志文件。此外，数据库提供schema架构一层用于数据库对象的管理。



图 2数据库文件布局

主要供自己复习参考使用，如有疏漏，深表歉意。

参考资料：

1. (美)本咁. SQL Server 2012 T-SQL基础教程[M]. 北京:人民邮电出版社, 2013.