1. /\*\*/注释多行， --注释单行。
2. 聚集索引。表数据是按聚集索引的顺序来存储的，一张表只能有一个聚集索引。非聚集索引是单独的结构，与表数据存储顺序无关。
3. GO不加分行，把代码分为不同的批次。
4. 在不符合命名规则的表名、列名加[]来使用它们.
5. WHERE子句的计算结果有true、false、unknown. 仅返回结果为true的行。
6. SELECT <column1>,<column2> FROM <schema>.<table> WHERE <column> [NOT] BETWEEN <value1> AND <value2>; [value1,value2]
7. SELECT <column1>,<column2> FROM <schema>.<table> WHERE <column> [NOT] LIKE 'value[a-c]' 'value[a,b,c]' value[^d]; %代表任意多个字符。\_代表一个字符。[]限制可选的一个字符.^限制不想要的一个字符
8. WHERE子句中AND优先级高于OR.
9. SELECT <column1>,<column2> FROM <schema>.<table> WHERE <column> [NOT] IN (<value1>,<value2>);
10. SELECT <column1>,<column2> FROM <schema>.<table> WHERE <column> IS [NOT] NULL;
11. SELECT <column1>,<column2> FROM <schema>.<table> ORDER BY <column1>[<sort direction>],<column2> [<sort direction>];
12. The concatenation operator (+) allows you to add together two strings. The syntax is simple: <string or column name> + <string or column name>.当+连接一个数字和一个字符串时，会将字符串转换为数字，如果字符串不能转换为数字，将出错。
13. ISNULL(value, replacement)如果value是null，则返回replacement。COALESCE(value1,value2,value3,……)返回第一个非null值。COALESCE是ANSI标准。
14. CAST和CONVERT进行类型转换。CAST(<value> AS <new data type>) CONVERT(<new data type>,<value>). CAST是ANSI标准。
15. 当两个不同优先级的数据进行操作时，结果类型为优先级高的那一个。
16. LEFT(string, <number of character>) RIGHT(string, <number of character>) get the specify number of character from the left or right side of string.
17. LEN(<string>)字符串中的字符数，DATALENGTH(<STRING>)返回字符串中的字节数。