**看视频**

可视化钻取

查询编辑器-高级编辑器-M语言显示

箱-数值型 列表-文本型

DAX-动态值。

智能表格-更改函数

想要在导入数据之后关联不同表格，可以直接拖动

在**参数**的**常规**里，可以改变图标的长度，宽度，仪表盘绝对位置，还可以在**参数**中改变**背景色**，**边框**，**标题**（颜色，文本）。**数据标签**可以显示值

**筛选器**中，**报告级**对所有页面都起作用。**页面级**只对本页面起作用。

点击表格的右上角，可以对图标的**计数**进行排序

**文本框**可以居中对齐。可视化的**卡片图**可以显示数值。

有的图表有分析，有的没有

发布后，可以保存为.pbix格式。可以选择手机布局。

**文章**

**转换**-**逆透视其他列**

**矩阵表**，**圆环图**，**切片器**，**堆积面积图**

Power Query中添加列有四种形式，重复列、索引列、条件列、自定义列：

**(一)添加重复列**

重复列就是把选中的列复制一列，以便对该列的数据进行处理而不损坏原有列的数据，

**(二)添加索引列**

索引列就是为每行增加个序号，记录每一行所在的位置，可以从0或者1开始，

**(三)添加条件列**

添加一列根据指定条件从其他列计算的数据，打开窗口看看，其实就是 if 函数，

**(四)添加自定义列**

自定义列就是使用M函数生成新的一列，

**追加查询**

追加查询是在现有记录的基础上，在下边添加新的行数据，是一种纵向合并，比如有两个表格式相同，需要合并为一个表，点击“追加查询”，

**合并查询**

如果说追加查询是纵向合并，那么合并查询就是横向合并，相当于Excel的VLOOKUP功能，就是匹配其他表格中的数据，不过PQ中的合并查询要比VLOOKUP功能强大的多，并且操作也更简单。

**标题提升**：**将第一行作为标题**。**将标题作为第一行**。

M函数

* M函数对大小写敏感，每一个字母必须按函数规范书写，第一个字母都是大写
* 表被称为Table，每行的内容是一个Record，每列的内容是一个List
* 行标用大括号{ }，比如取第一行的内容：=表{0} //PQ的第一行从0开始
* 列标用中括号[ ]，比如取自定义列的内容：=表[自定义]
* 取第一行自定义列的内容：=表{0}[自定义]

常用的M函数

**聚合函数：**

求和：List.Sum()

求最小值：List.Min()

求最大值：List.Max()

求平均值：List.Average()

**文本函数：**

求文本长度：Text.Length()

去文本空格：Text.Trim()

取前n个字符：Text.Start(文本,n)

取后n个字符：Text.End(文本,n)

**提取数据函数：**

从Excel表中提取数据：Excel.Workbook()

从Csv/Txt中提取数据：Csv.Document()

**条件函数：**

if else then （相当于Excel中的IF)

从哪里查找M函数

新建一个空查询，在公式标记栏中输入=#shared，就把所有的M函数显示出来了，点击某个函数，最下方便出现该函数的注释：

基数

基数就是两个连接字段的对应关系，分为多对一、一对一和一对多，一对多和多对一其实是一样的，实际上就是两种关系：

多对一（\*：1）：这是最常见的类型，代表一个表中的关系列有重复值，而在另一个表中是单一值

一对一（1：1）：两个表是一对一的关系，列中的每个值在两个表中都是唯一的

具有唯一值的表通常称为“查找表”，而具有多个值的表称为“引用表”。在上述的关系图上，产品明细表上类别手机、平板、电脑都不是唯一的，每个品牌都有这种类型，是个引用表；但类别表上，几种类别都是唯一值，因此这两个表是多对一的关系，类别表也就是查找表。

交叉筛选方向

表示数据筛选的流向，有两种类型：

双向：两个表可以互相筛选

单向：一个表只能对另一个表筛选，而不能反向

这个稍微有点抽象，以后可以根据实例来理解。

DAX函数

DAX是英文Data Analysis Expression的缩写，意思是数据分析表达式，从名称上就可以看出，DAX公式是用作数据分析的，事实上也确实如此，从数据分析层面认识DAX会更有助于我们理解它.

DAX参数的基本格式

* **表名用单引号' '括着 // '日期表'**
* **字段用中括号[ ]括着 //[日期]**
* **度量值也是用中括号[ ]**
* **引用字段始终要包含表名，以和度量值区分开**
* **还有””**

DAX常用函数

**聚合函数**

* SUM
* AVERAGE
* MIN
* MAX

这几个函数的使用和在Excel中完全一致，DAX中还有一类特有的函数非常有用，和这几个函数很像，后面加个X，

* SUMX
* AVERAGEX
* MINX
* MAXX
* RANKX

这几个函数可以循环访问表的每一行，并执行计算，所以也被称为迭代函数。

**其他常见的聚合函数：**

* COUNT:计数
* COUNTROWS：计算行数
* DISTINCTCOUNT:计算不重复值的个数

**时间智能函数**

* PREVIOUSYEAR/Q/M/D:上一年/季/月/日
* NEXTYEAR/Q/M/D:下一年/季/月/日
* TOTALYTD/QTD/MTD:年/季/月初至今
* SAMEPERIODLASTYEAR:上年同期
* PARALLELPERIOD:上一期
* DATESINPERIOD:指定期间的日期

关于时间智能函数，会专门写篇文章介绍，利用它可以灵活的筛选出一段我们需要的时间区间，做同比、环比、滚动预测、移动平均等数据分析时，都会用到这类函数。

**筛选函数**

FILTER:筛选

ALL:所有值，可以清除筛选

ALLEXCEPT:保留指定列

VALUES:返回不重复值

这几个函数，就是典型的DAX查询函数，通过筛选来操纵上下文的范围。

这就是度量值的魅力，只是简单建立了四个度量值，然后就神奇的实现了多维度的比较，这要归功于度量值的最重要的特征：上下文。

上下文就是度量值所处的环境，筛选表的行列标签、切片器的选中，都是度量值的上下文，比如北京2017年截至5月的苹果手机累计销售额3424000，它的上下文就是下面这5个维度：

* [城市]="北京市"
* [品牌]="苹果"
* [类别]="手机"
* [年度]=2017
* [月份]=5月

理解了上下文，就理解了度量值的最重要的特征，因此她被称为**移动的公式**，随着上下文环境的不同而展示不同的数据，度量值另外还有两个主要特征：

**度量值不浪费内存**，只有被拖到图表上才执行运算，如果数据量非常大的时候这点非常有利

**度量值可以循环使用**，比如上面的建立的度量值：

本年累计销售额 = TOTALYTD([销售总额],'日期表'[日期])

累计同比增长率 = divide([本年累计销售额],[上年累计销售额])-1,

就是直接调用之前建立好的度量值，**所以以后在模型中新建度量值的时候，推荐从最简单的度量值开始建。**

示例

本年累计销售额 = TOTALYTD([销售总额],'日期表'[日期])

上年累计销售额 = TOTALYTD([销售总额],SAMEPERIODLASTYEAR('日期表'[日期]))

累计同比增长率 = divide([本年累计销售额],[上年累计销售额])-1

月份 = format('日期表'[日期],"MM")

CALCULATE的语法结构

语法：

CALCULATE(<expression>,<filter1>,<filter2>…)

第一个参数是计算表达式，可以执行各种聚合运算

从第二个参数开始，是一系列筛选条件，可以为空；如果多个筛选条件，用逗号分隔

所有晒选条件的交集形成最终的筛选数据集合

根据筛选出的数据集合执行第一个参数的聚合运算并返回运算结果

通过从第二个参数开始的筛选条件，得到一个数据集合，并利用第一个参数执行聚合运算，这不就是DAX要实现的功能：提取有用数据并执行聚合运算吗，所以说CALCULATE几乎就是DAX本身，它就是实现DAX功能的引擎，并能灵活的操控外部上下文，后面的数据分析也都离不开CALCULATE的身影。

另外，介绍CALCULATE语法的时候说，第一个参数为聚合运算表达式，为什么这里没有用聚合函数而只用了一个度量值呢？这是因为度量值[产品数量]本身就是一个聚合函数运算，实际上度量值[产品数量1]等同于这个：

产品数量 = COUNTROWS('产品明细')

产品数量1 = CALCULATE([产品数量])

产品数量1 = CALCULATE(COUNTROWS('产品明细'))

DAX函数可以直接引用已经创建好的度量值，可以使DAX函数看起来更简洁、更具可读性，这也是建议从最简单的度量值开始建的原因。

filter函数

FILTER语法

FILTER(<table>,<filter>)

* 第一个参数<table>是要筛选的表
* 第二个参数<filter>是筛选条件
* 返回的是一张表，不能单独使用，需要与其他函数结合使用

问题：<https://zhuanlan.zhihu.com/p/64383000>

这两个数字正是内部上下文和外部上下文相结合，筛选计算的结果