

Power BI pdf

- 界面认识

- 常用：获取数据、新建度量值、新建列、新建表和右边的筛选器、可视化模块



- powerBI中各图表适用数据

- 选图思考

- 1、明确问题类型

- 比较：比较多少/谁最少：条形/柱状/瀑布/表格
- 组成：占比如何/随时间是否改变：堆积柱/饼图/树状图/堆积面积
- 趋势：随时间如何演化：折线/面积/移动平均/季节分解
- 分布：数据如何分散？是否偏态/有离群？：直方图、箱线图、核密度（或小提琴图）
- 关系：变量间相关/模式？：散点图、气泡图、热力图、相关矩阵
- 地理：空间分布/热区？：地图/热力图/气泡地图
- 流程/漏斗：流转/转化率？：漏斗图、桑基图
- 异常检测 / 分解：哪个因素导致变化？：瀑布、分解树、帕累托图

■ 2、看数据类型

- 名义（男/女）：条形、饼、树图
- 序数（学位排名）：条形（按序排序）、箱线（分位）等
- 区间/比率：折线、直方图、散点、箱线
- 时间：折线、面积、堆积时间序列
- 地理：地图类

■ 3、维度VS指标

- 维度：通常放在轴/图例（如产品、地区、月份）
- 指标：聚合值（如销售额、库存天数、毛利率）放在数值轴/大小/颜色

■ 4、聚合与粒度一致性

- 始终确保所选图表的时间/粒度一致（天/周/月/年），并注明聚合方式

○ 常见图表对照

■ 趋势与时序

- 折线图
 - 适合：连续时间序列（天/周/月/年），单条或多条（多类别比较）
 - 聚合：Sum(Sales) by Month(Date); Avg(Margin) by Quarter
 - 回答：销售随时间如何变化？季节性/趋势/拐点在哪？
 - 优点：清晰显示趋势与拐点；支持移动平均
 - 实现要点：处理缺失日期、平滑（移动平均）或季节分解；多序列时区分颜色/图例
- 面积图 / 堆积面积
 - 适合：展示总量随时间与其构成的变化（随时间的组成）
 - 聚合：Stacked Sum(Sales) by Product & Month
 - 回答：总体增长怎样？各产品线占比如何随时间改变？

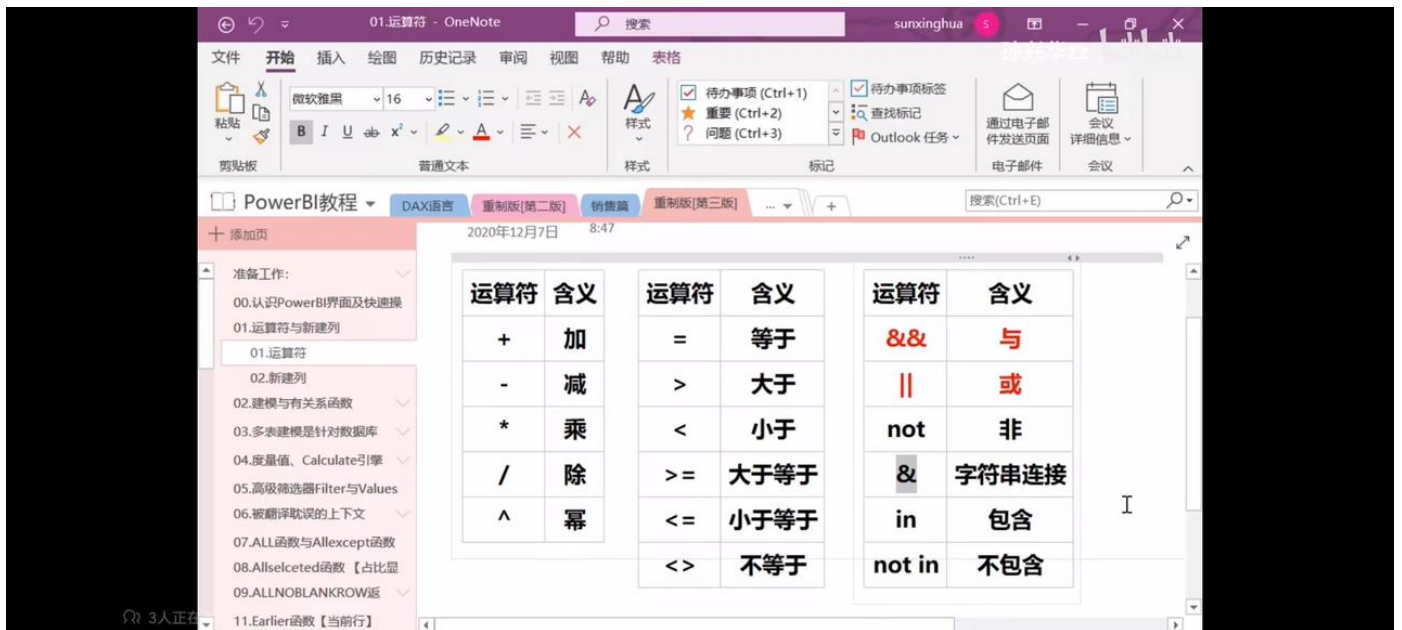
- 缺点：堆叠时底层曲线不易比较，适合关注整体与部分趋势
- 折线 + 柱状
 - 适合：需要同时展示不同量纲（量 vs 比率），如销售额（柱）与毛利率（线）
 - 优点：对比两类指标便捷
 - 实现要点：使用次坐标轴需谨慎，标注单位
- 比较（类别间）
 - 垂直/水平柱状图
 - 适合：对比不同类别（产品、地区、供应商）的大小
 - 聚合：Sum(Sales) by Product
 - 回答：哪些产品销量最高？排名如何？
 - 优点：直观、易读；水平条适合长文本类目
 - 实现要点：按值降序排序、显示数据标签、限制类别数量
 - 堆叠/簇状柱状图
 - 适合：比较并展示组成（不同类别内部结构）
 - 回答：每个地区的销售由哪些产品贡献？
 - 注意：堆叠会遮盖底层细节，适合看总体构成
 - 瀑布图
 - 适合：展示从初始值到最终值的逐步变化（新增/减少项）
 - 示例：Gross Profit 分解：Revenue → COGS → OpEx → Tax → Net Income
 - 回答：哪些项对变化贡献最大（正/负）？
 - 优点：解释增减来源，适合财务分解
 - 实现要点：正确标记中间小计与总计，排序合理
 - 帕累托图（Pareto：柱 + 累积折线）
 - 适合：识别少数关键项（80/20 原则）
 - 示例：按供应商计数的缺陷数量，找Top contributors
 - 回答：哪些少数类别造成大部分问题？
 - 实现要点：同时显示柱（值）和累积百分比线
- 组成 / 占比
 - 饼图 / 圆环图
 - 适合：显示单一时间点的占比（类别数量少时）
 - 回答：市场份额/品类占比是多少？

- 缺点：类别多或差异小不易比较；尽量限制最多 4-6 个切片
- 实现要点：使用标签与百分比，考虑替代（条形图）以便比较
- 树形图
 - 适合：层级结构的部分—整体展示（产品->子产品->SKU）
 - 回答：哪个子类贡献最大？层级内的相对大小
 - 优点：能显示多层级；适合大量类别
 - 缺点：相对精确度差，难以比较相近矩形
- 堆积条形图
 - 类似堆积柱，但适用于类别名长或横向阅读更佳
- 分布与离群点
 - 直方图
 - 适合：数值型单变量分布（价格、订单金额、交付天数）
 - 回答：数据是否偏态？是否有多峰？
 - 实现要点：选择合适的 bin 数（规则：Sturges、Freedman–Diaconis 等），Power BI 中需要自定义 bin
 - 箱线图
 - 适合：展示分位数、离群值（库存天数在区域/产品间比较）
 - 回答：哪类产品库存波动最大？是否有异常值？
 - 优点：对比多组分布非常有效
 - 实现要点：需注意样本量（小样本箱线不稳定）
 - 小提琴图
 - 适合：显示密度分布（更细腻的分布形态），Power BI 可通过自定义可视化实现
- 关系与相关
 - 散点图
 - 适合：两个连续变量关系
 - 回答：是否有相关性、线性/非线性模式？
 - 实现要点：可加回归线、点大小/颜色编码第三变量（如盈利率）
 - 气泡图
 - 适合：三变量（x, y, size）关系，如 Region Sales (x), Growth (y), Market Share (size)
 - 注意：气泡面积不能直接与数值线性对应，图例说明单位

- 热力图 / 矩阵
 - 适合：两个分类变量交叉的值强弱（Product × Month 的销量强度）
 - 回答：在哪些月份/产品组合表现强/弱？
 - 实现要点：颜色尺度选择（连续 vs 分类），配合数据标签。
- 地理与位置信息
 - 气泡地图 / 填充地图
 - 适合：按地区/国家/城市展示指标
 - 回答：哪些区域表现强？哪里潜力大？
 - 实现要点：处理地名歧义（标准化）、按人口或门店数做归一化
 - 热力地图
 - 适合：局部密度或热度（线下门店热度、订单密度）
 - 实现要点：需要经纬度或地图服务支持
- 流程 / 漏斗 / 关联
 - 漏斗图
 - 适合：展示转化步骤（访客→注册→下单→复购）
 - 回答：在哪一步转化率掉得最多？
 - 实现要点：明确步骤顺序与样本口径
 - 桑基图
 - 适合：展示流量在类别间如何流转（渠道→产品类→地区）
 - 优点：直观显示流向与流量大小
 - 实现要点：需要构建流向数据
 - 网络图 / 力导向图
 - 适合：展示关系网（供应商-零件-工厂）、客户推荐网络
 - 注意：复杂度高，适合探索而非高层报告
- 指标卡片与概览
 - KPI 卡 / 卡片
 - 适合：显示关键指标
 - 用法：常放在仪表盘顶部，搭配趋势箭头与环比/同比数值
 - 仪表 / 量规
 - 适合：显示与目标或阈值的关系（达成率）
 - 注意：不适合精确比较多个项，适合单一目标展示

- 解释性 / 根因分析
 - 瀑布（见上） & 分解树
 - 分位/箱线组合 & 交互式筛选
 - 用于做假设检验、场景分析与灵敏度分析
- 为财务 / 存货 / 运营 场景给出精选搭配
 - 财务月度看板
 - KPI 卡：Total Revenue, Net Profit, Operating Margin, Cash Balance
 - 折线图：Revenue & Net Profit by Month（含 YoY）
 - 瀑布图：从 Revenue 到 Net Income 的分解（展示主要增减项）
 - 帕累托/条形：按客户/产品的收入贡献 Top10
 - 箱线/直方图：单笔交易金额分布（检测异常）
 - 地图（若跨区）：按地区 Revenue per Store（归一化）
 - 存货/供应链看板
 - 条形：库存按 SKU 排序（Top N）
 - 饼/树图：库存按类别占比
 - 折线：库存天数（Days of Inventory）月度趋势
 - 散点：库存量 vs 销售速度（识别滞销/超储）
 - 热力图：仓库 × SKU 的库存密度（补货优先级）
 - 漏斗/流程：采购到入库的周期漏斗（识别瓶颈）
 - 运营/销售看板
 - 漏斗图：Lead → MQL → SQL → Opportunity → Closed
 - 气泡图：Region Sales vs Growth vs Market Share
 - 地图：门店表现与覆盖密度
 - 分解树：找出影响 Top 销售产品的维度（渠道、促销、价格区间）
 - 时间序列：促销期间 vs 非促销期间销量对比（折线 + 区域阴影）
- 选图速查流程（3 步决策）
 - 我要回答的问题是什么？（比较 / 趋势 / 分布 / 关系 / 地理 / 流程）
 - 我的主变量是什么 & 粒度（时间/类别/数值）？
 - 是否需要展示构成（部分→整体）或解释增减来源？
 - 类别数（少→饼/柱，多→树/条）、时间序列优先折线、关系用散点、分布用直方/箱线
- 实用设计与实践建议

- 使用 切片器 (Slicers) 提供交互 (时间、产品、地区)
- 对于 Top N 显示: 用 DAX 做 TopN + “Others” 合并, 避免图表拥挤
- 色彩: 保持一致含义 (例如红表示负, 绿色正); 避免过多颜色
- 轴与单位: 标注单位 (K/M), 使用格式化 (千/百万)
- 注释与解释: 关键拐点添加文本框 (Insight annotation)
- 交互: 利用 Drillthrough / Decomposition Tree 给出 “下一层” 洞察
- 性能: 大量点用聚合/抽样 (scatter over thousands of points 会卡)
- 可访问性: 为色盲用户选择色板, 确保标签可读
- 行动建议
 - 明确每个报表的 核心问题 (1句话), 然后只保留回答该问题的图表
 - 每页仪表盘控制在 3-6 个可视化 (避免认知过载)
 - 对于探索性分析 (分析师视图) 使用更多交互式图 (散点、分解树、Sankey); 对高层管理用 KPI + 趋势 + 瀑布
 - 对金融/库存类数据, 强烈推荐: 瀑布 (财务分解) + 箱线/直方 (分布/异常) + 堆积时间序列 (构成趋势) 这三种组合模版
- 基础知识
 - 运算符



- 新建列
 - 定义列名=对其他列进行计算得出该列的值（[]代表选这一列，‘代表选某一张表的某一列）
 - 注意如果要跨表计算，那就要先给两张表建立关系
- 建模
 - 导入多张表就要先看建模（比如销售明细与商品表）
 - 销售明细可以向商品表索取内容，商品表可以筛选销售明细
 - lookupvalue（把哪张表的哪个列拿过来，匹配那张表的谁，找自己表里的谁）
 - 比如：lookupvalue=（商品表进价列，商品表品名列，销售明细商品名称）
- 关系函数

- 销售明细表索取商品表内容（多端索取1端，正常的vlookup）
 - 比如：销售成本=[销售数量]*related（'商品表'[进价]）
 - 多端提取1端的时候用related
- 商品表筛选销售民系表（1端筛选多端，类似于数据透视）
 - 比如：订单数量=countrows（relatedable('销售表'））
 - 1端筛选多端的时候用relatedable
- 度量值
 - 计算列的总计
 - 新建度量值：总销量=sum(变量)
 - 然后可以把“总销量”放在卡片，可以做成一个KPI值
- calculate筛选引擎
 - 单条件（单列）
 - 比如说我只想要总销量里面类别为A的：
 - A产品销量1=calculate([总销量]，‘商品表’[品名]=“A”)
 - A产品销量2=calculate（[总销量]，‘销售表[品名]=“A”’）
 - 以上两种有什么不同？如果矩阵图表中用的销售表的品名，那么calculate计算公式就应该用商品表的品名；如果矩阵图表中用的商品表的品名，那么calculate计算公式就应该用销售表的品名。别让他们来自同一张表就可以了。
 - 同表过滤（Single-table Filtering）导致无法穿透到另一张表，从而 measure 不生效或结果错误
 - 多条件（多列）
 - 多条件=calculate（[总销量],商品表'[品名]=“A”，‘商品表’[进价]=0.1）
 - 多条件1=calculate([总销量]，‘商品表’[品名]in{"A""B""C"})
 - 多条件2=calculate([总销量]，not ‘商品表’[品名] in{"A""B""C"})
 - 文字字符必须打引号，数字不需要
- 筛选表
 - 比如说我有一张销售表，我想产生一张新的销售表，而且我想通过别的表里的条件去筛选他
 - 点击新建表，新表=calculateable('销售表'，‘商品表’[品名]=“A”,'商品表'[进价]=0.1)
- 度量值管理
 - 主页-输入数据-命名-加载-在这个空表中建度量值
- 高级筛选器filter

- 是一个表函数，可以将filter公式直接新建表测试
- 把某张表以什么条件进行筛选，不能单独使用
- filter函数对筛选的表进行横向逐项扫描，这样的函数叫迭代函数
- countrows(filter(表, 筛选条件))
- calculate (表达式 (或度量值), filter (‘表名’, 筛选条件))
- 正常情况下也可以用calculate筛选，适用于列VS固定值时，但是当列VS度量值、列VS公式、列VS列、度量值VS度量值、度量值VS公式、度量值VS固定值时，需要用filter
- 比如：大于250分的人的平均分
- filter有个要求：他要求用在filter里面的表必须是唯一值的表（比如说花名册、商品表），不能是多端的表（比如成绩表、销售明细表），如果没有唯一值的表，就只能人工造表
- values人工造表（相当于去除重复值）
 - 新建表：测试表=values (成绩表[学号])，这样就形成了一个唯一值的表，然后用于filter里面
- 上下文
 - 新建列时行上下文（逐行扫描，就是把这个值求出来之后，每一行都显示这个值），度量值是筛选上下文
 - 度量值天生就有筛选功能，新建列是行上下文，行上下文没有筛选功能，想让行上下文实现筛选功能就要在外面套一个calculate
- all函数
 - 清除度量值的筛选功能，因为度量值天生带有筛选功能
 - 主要是用于计算占比，要有一列不能筛选的总数=countrows(all(‘商品表’))，表示的就是商品表总数量，而且不会被其他度量值影响
 - all函数有两种语法：all(表);all(表[列])
 - all(表)主要用于计算唯一值行数总数，当all函数为表时，忽略所有筛选条件，无论是外部切片还是内部
 - all(表[列])主要用于计算比如某一系列数量合计
- 聚合函数
 - sum (表[列])
 - average (表[列])
 - average 只统计数字 (1+2+3) /3
 - averagea 除所有项数 (1+2+3) /4
 - 最大值/最小值

- countrows (表) 记录数
- distinctcount(表[列])去重之后的计数
- counta:统计非空数
- countblank统计空白数量
- product:计算列中单元格乘积
- 迭代函数
 - 对每一行逐行扫描
 - 就是聚合函数后面带了个X
 - 新建列
 - 要加calculate, 不然每一行的结果都是最后的总数
 - sumx(表, A*B), averagex、rankx、minx/maxx
- Values和Distinct的区别 (表函数)
 - Values(表[列]): 提取唯一值
 - Values(表): 复制表
 - Distinct(表[列]): 提取唯一值
 - Distinct(表): 删除重复行 (每列都一样的去除)
 - Distinct(Filter(表, 表[列]= “某条件”))这样就可以取到一个在某条件下去重的表
- 条件判断函数
 - IFERROR遇到错误时指定数值替换
 - 切记: 新建列是行上下文, 是逐行扫描的, 一定要加calculate
 - IFERROR ([销售金额]*[单价],BLANK())
 - IF条件判断
 - IF ([性别]=男, 先生, 女士)
 - IF ([取款日期]=BLANK(),BLANK(),[取款日期]-[存款日期])
 - SWICH多条件判断
 - 月份=swich(表[列], 1, “一月”, 2, “二月”, 3, “三月”, 4, “四月” ...,未能识别)
- 安全除法DIVIDE
 - DIVIDE (分子, 分母, {替换结果})
- 按层级计算ISINSCOPE (分层求占比)
 - 先做一个度量值: SUM

- 占比层级=SWICH (TRUE(),
 - ISINSCOPE(商品表[商品名称]), DIVIDE([总金额],CALCULATE([总金额], ALLSELECTED(商品表[商品名称]))) ,
 - ISINSCOPE(商品表[产品类别]), DIVIDE([总金额],CALCULATE([总金额], ALLSELECTED(商品表[产品类别]))) ,
 - DIVIDE([总金额], CALCULATE([总金额], ALLSELECTED(商品表[商品名称])))
- 如果商品名称有层级关系, 怎么怎么做, 如果产品类别有层级关系, 怎么怎么做
- firstnonblank和lastnonblank
 - 可以当表函数也可以当值函数
 - firstnonblank(列, 表达式)
 - 应用于表函数
 - 表1=firstnonblank (表[日期], CALCULATE (SUM (表[销售])))
 - 产生第一笔销售额的日期
 - 应用于值函数
 - 计算每个人不为空的销售
 - SUM
 - firstnonblank (表[日期], CALCULATE (SUM (表[销售])))
- HASONEFILTER
 - 语法: HASONEFILTER (列名)
 - 当指定列有且只有一个由直接筛选产生的值时, 返回true, 必须使用物理列, 不能是表达式
- HASONEVALUE
- DATATABLE人工建表
 - DATATABLE(字段名1, STRING, 字段名2, STRING{{数据11, 数据12}, {数据21, 数据22}})
- 日期表
 - 日期表 = ADDCOLUMNS(CALENDAR(date(2024,02,10),date(2025,02,10)), "年", YEAR ([Date]),
 - "季度", ROUNDUP(MONTH([Date])/3,0), "月", MONTH([Date]), "周", weeknum([Date]),
 - "年季度", year([date]) & "Q" & ROUNDUP(MONTH([Date])/3,0),
 - "年月", year([Date]) * 100 + MONTH([Date]), "年周", year([Date]) * 100 + weeknum([Date]), "星期几", WEEKDAY([Date]))
- selectedvalue

- 当指定列中只有一个值时返回该值，否则返回备选结果，省略备选结果时返回空值
- selectedvalue(列, 备选值)
- 如果需要做切片器，可以创建一张表，利用创建表的字段名做切片器
- 转换函数
- CURRENCY数字转货币
 - 新建列
- INT向下舍入
- TRUNC直接取整
- ROUND四舍五入
- MROUND取数值的倍数
 - 两个参数符号必须一致
- 日期和时间的转换
 - date()
 - time()
- CONVERT转换为指定数据类型
 - CONVERT（表达式，数据类型）
 - INTERER整型，DOUBLE双精度，STRING字符，BOOLEAN布尔，CURRENCY货币，DATETIME日期时间
 - TIMEVALUE将文本格式转换成日期时间格式
 - VALUE将文本转数值
- FORMAT设置成指定的数字格式
 - FORMAT(DATE([年],[月],[日]),"yyyy年mm月dd日")
 - 通常于DATE和TIME一起用
 - 新建列
 - 1

附2: 预定义的数字格式 - OneNote

搜索

sunxinghua

文件 开始 插入 绘图 历史记录 审阅 视图 帮助

剪贴板

格式刷

普通文本

样式

标记

电子邮件

PowerBI教程

DAX语言 重制版[第二版] 销售篇 商品篇 重制版[第三版]

搜索(Ctrl+E)

添加页

11.Values与Distinct区别

12.条件判断函数[基础]

13.安全除法DIVIDE与按层

14.ISCROSSFILTERED函数

15.FIRSTNONBLANK与LAST

16.进阶条件判断函数

17.转换函数

18.FORMAT函数【格式化】

附1: 日期格式

附2: 预定义的数字格式

附3: 自定义数字格式

19.日期时间函数【非智能】

19.文本函数

General Number

显示没有千分符的数字

Currency

显示带千分符的数字, 对于符合要求的数字, 在十进制分隔符的右边显示两个数字。输出的内容基于系统区域设置。

Fixed

在十进制分隔符的左边显示至少一个数字, 右边显示至少两个数字。

Standard

显示带有千分符的数字, 小数点分隔符的左边至少有一个数字, 右边至少有两个数字。

Percent

将显示的数字乘以 100, 并在右侧添加百分号(%); 始终在十进制分隔符的右边显示两个数字。

Scientific

使用标准科学符号, 提供两个有效数字。

Yes/No

如果数字为 0, 则显示 No; 否则显示 Yes。

True/False

如果数字为 0, 则显示 False; 否则显示 True。

On/Off

如果数字为 0, 则显示 Off, 否则显示 On。

FORMAT(12345.67, "General Number") = 12345.67

FORMAT(12345.67, "Currency") = ¥12345.67

FORMAT(12345.67, "Fixed") = 12345.67

FORMAT(12345.67, "Standard") = 12,345.67

FORMAT(12345.67, "Percent") = 1,234,567.00 %

FORMAT(12345.67, "Scientific") = 1.23E+04

1人正在

附1: 日期格式 - OneNote

搜索

sunxinghua

文件 开始 插入 绘图 历史记录 审阅 视图 帮助 表格

剪切 复制 粘贴 格式刷

等线 16 普通文本

样式

文章来源 (Ctrl+1) 博客题材 (Ctrl+2) 与<人员 A> 讨论 (Ctrl+3) 待办事项标签 查找标记 Outlook 任务 通过电子邮件发送页面 会议 详细会议信息

PowerBI教程

DAX语言 重制版(第二版) 销售篇 商品篇 重制版(第三版)

搜索(Ctrl+E)

添加页

11.Values与Distinct区别

12.条件判断函数(基础)

13.安全除法DIVIDE与按层

14.ISCROSSFILTERED函数

15.FIRSTNONBLANK与LAST

16.进阶条件判断函数

17.转换函数

18.FORMAT函数【格式化】

附1: 日期格式

附2: 预定义的数字格式

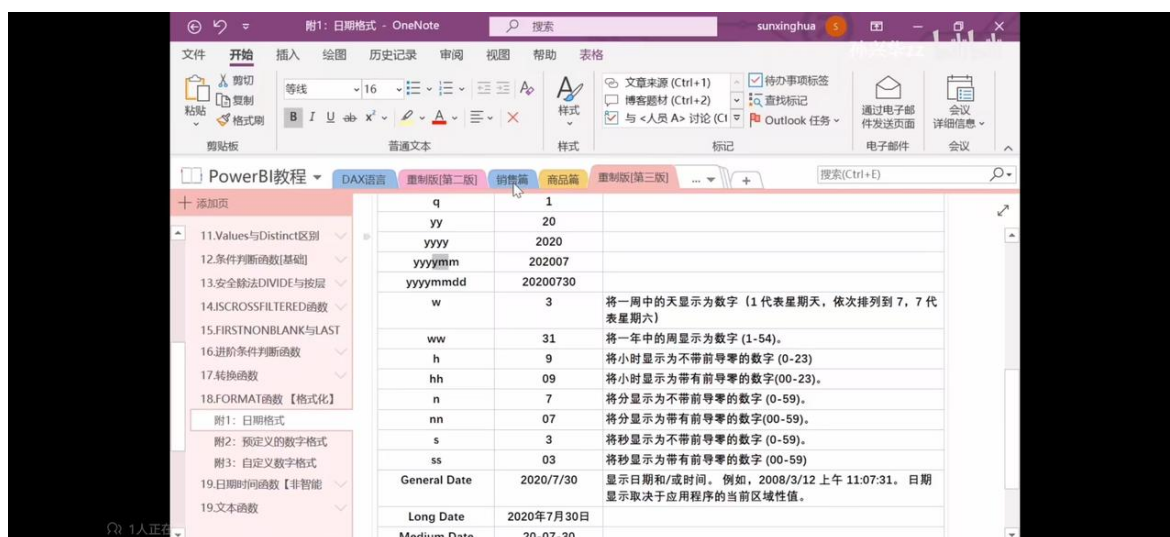
附3: 自定义数字格式

19.日期时间函数【非智能】

19.文本函数

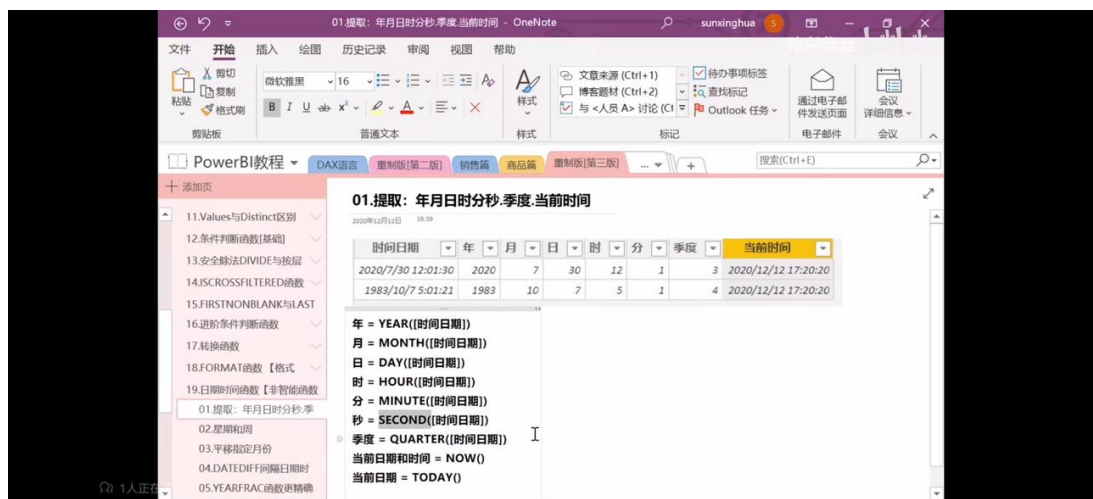
d	1
dd	01
ddd	Mon
dddd	Monday
ddddd	1980/7/30
ddddd	1980年7月30日
aa	周一
aaaa	星期一
m	1
mm	01
mmm	Jan
mmm	January
OO	1月
OOO	一月
q	1
yy	20
yyyy	2020

1人正在



时间日期函数

公式



- 注意周（因为是从周日开始的）

- weekday([时间日期], 2)
- weeknum([时间日期], 2)

- 平移指定月份

- EDATE([日期], 平移月数)
- EOMONTH (TODAY(),-1) 代表今天的日期向前推一个月的月末那一天

- DATEIFF间隔日期时间

- DATEIFF(起始日期, 结束日期, 间隔单位（下拉可选）)

- yearfrac算工龄

- 这个也是计算两个日期之间的间隔，但是这个间隔不用具体的年月天来表示，而是通过相隔比例来表示
- yearfrac([起始日期], TODAY(),0)

- 文本函数

- CONCATENATE将两个字符串连接
- 连接文本=CONCATENATE([文本1],[文本2])
- 连接数字=CONCATENATE([数字1],[数字2])
- 新建列
- 一般配合if或switch
- FIND找字符串在另一个字符串的起始位置（区分大小写，不支持通配符）
- SEARCH同上（不区分大小写，支持通配符）
- 截取函数
 - left,right,mid,len
- fixed数值转字符按指定小数位四舍五入
- 大小写转换
 - lower, upper
- 删除空格
 - TRIM
 - 配合截取函数
- 重复字符串
 - REPT（文本，重复次数）
- 替换函数replaced
 - 按位置替换
 - 按字符替换
- 替换函数SUBSTITUTE（区分大小写）
- 三角函数、数学函数、信息函数、取余数MOD（不常用）
- 常用公式汇总：
 - 1

一、基础聚合函数

1、SUM

对数值列求和

总销售额 = SUM('销售表'[金额])

2、AVERAGE

计算数值列的平均值（自动忽略非数值）

平均单价 = AVERAGE('产品表'[单价])

3、COUNT

统计非空值的行数

订单数 = COUNT('订单表'[订单ID])

4、DISTINCTCOUNT

计算列中唯一值的数量

客户数 = DISTINCTCOUNT('客户表'[客户ID])

5、MIN/MAX

返回列的最小值/最大值

最低温度 = MIN('天气表'[温度])

二、筛选与上下文控制

1、CALCULATE

在指定条件下计算表达式（DAX核心函数）

华东销售额 = CALCULATE([总销售额], '区域表'[地区]="华东")

2、FILTER

返回满足条件的表（需搭配CALCULATE）

高额订单 = CALCULATE([总销售额], FILTER('订单表', [金额]>10000))

3、ALL

忽略当前筛选上下文

占比 = DIVIDE([销售额], CALCULATE([销售额], ALL('产品表')))

4、ALLSELECTED

保留用户手动选择的筛选

累计占比 = DIVIDE([销售额], CALCULATE([销售额], ALLSELECTED('日期表')))

三、时间智能函数

1、DATEADD

日期偏移计算

上月销售额 = CALCULATE([销售额], DATEADD('日期表'[日期], -1, MONTH))

2、SAMEPERIODLASTYEAR

同比日期范围

同比销售额 = CALCULATE([销售额], SAMEPERIODLASTYEAR('日期表'[日期]))

3、DATESYTD

年初至今日期范围

YTD销售额 = CALCULATE([销售额], DATESYTD('日期表'[日期]))

4、OPENINGBALANCEMONTH

获取月初值（财务常用）

月初库存 = OPENINGBALANCEMONTH(SUM('库存表'[数量]), '日期表'[日期])

四、迭代函数（X函数）

1、SUMX

逐行计算后求和

总成本 = SUMX('订单表', [数量] * RELATED('产品表'[成本价]))

2、AVERAGEX

逐行计算后求平均

平均利润率 = AVERAGEX('销售表', [利润]/[销售额])

3、RANKX

动态排名（需搭配ALL）

销售排名 = RANKX(ALL('销售表'[销售员]), [销售额])

4、MAXX/MINX

返回表达式的最大/最小值

最高单价 = MAXX('产品表', [单价])

五、关系与跨表计算

1、RELATED

从关联表获取值（需存在关系）

产品类别 = RELATED('产品表'[类别])

2、LOOKUPVALUE

无关系时跨表查询

客户城市 = LOOKUPVALUE('客户表'[城市], '客户表'[ID],
[客户ID])

3、USERELATIONSHIP

临时激活非活动关系

发货销售额 = CALCULATE([销售额],
USERELATIONSHIP('日期表'[日期], '订单表'[发货日期]))

六、文本处理

1、CONCATENATEX

合并多行文本

客户列表 = CONCATENATEX(VALUES('客户表'[名称]), [名称], ", ")

2、FORMAT

格式化输出

格式化日期 = FORMAT(TODAY(), "yyyy年MM月")

3、SEARCH

查找文本位置

是否VIP = IF(SEARCH("VIP", [客户等级]) > 0, "是", "否")

七、错误处理

1、IFERROR

错误时返回替代值

安全除法 = IFERROR([销售额]/[数量], 0)

2、ISBLANK

检查是否为空值

是否缺货 = IF(ISBLANK([库存量]), "缺货", "有库存")

关键技巧：

1. CALCULATE是核心：

90%的高级分析依赖它修改筛选上下文。

2. 迭代函数性能低：

避免对大表使用SUMX/AVERAGEX。

3. 时间智能依赖日期表：

必须创建连续的日期表。