3.3 Notion 函数基础入门（一）：用四则运算理解 Formula 的基本特性与用法

> 前言我始终认为函数公式字段（Formula）是 Notion
> 最难入门、也是最难掌握的一部分内容，因为你几乎不能从其他任何地方继承关于
> Formula 的基础知识，所有语法和用法都需要从头开始学起。

前言

我始终认为函数公式字段（Formula）是 Notion
最难入门、也是最难掌握的一部分内容，因为你几乎不能从其他任何地方继承关于
Formula 的基础知识，所有语法和用法都需要从头开始学起。

虽然与正经的编程语言或者 Excel 的复杂公式相比，Notion Formula
的语法规则并不算深，但依然需要你首先理解各个函数的意义和用途，然后才能结合自己的需求去组合、应用它们，这里的每个步骤都存在一定的难度。

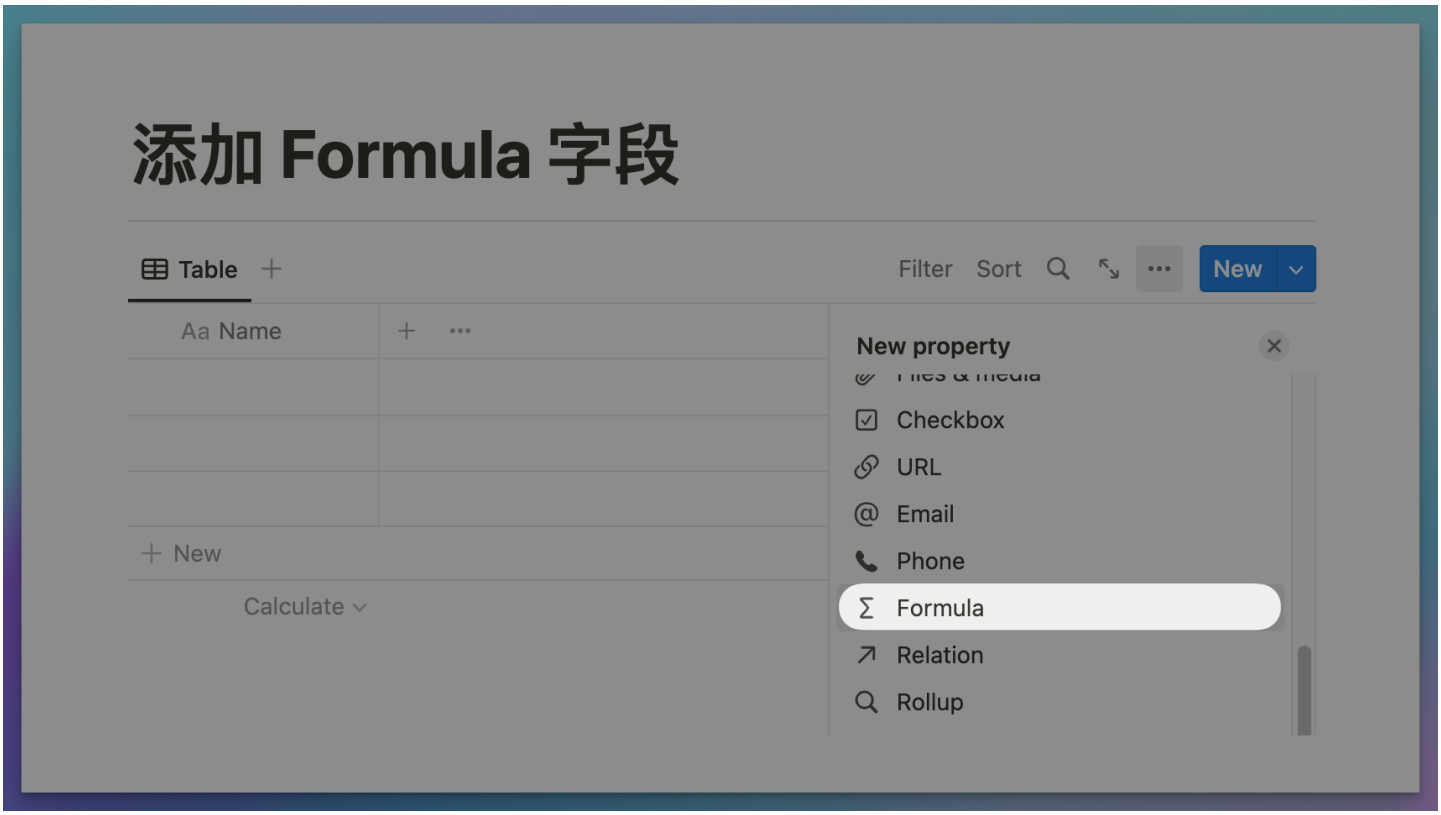
遗憾作者本人对 Formula
的理解和应用还处在
比较初级的水平，不足以向各位展示清楚它的全貌，仅能在此带新手朋友们稍作入门。诚然挑战艰巨，我还是会尽力讲清楚我已学会的每一个函数和用法。如果你有发现任何错漏的地方也欢迎在评论区留言反馈，我也会在第一时间订正。

需要注意的是，如果你发现学习 Notion
的公式是艰巨的、痛苦的、不仅对提高效率产出并无太大帮助，反而因此浪费了过多的时间的话，请不要犹豫就此打住吧。能学会
Formula 当然很好，但就算学不会，Notion
的魅力也并不会因此有所折损，它只是锦上添花的一种辅助。

另外也请不要把 Excel 的操作习惯带入到 Notion 函数中，Notion
的操作方法、功能上限与 Excel 相比都有非常大的差别。

Formula 基础用法

Notion 的公式函数只能创建在数据库中，并以「新建 Formula
字段」的方式来添加：

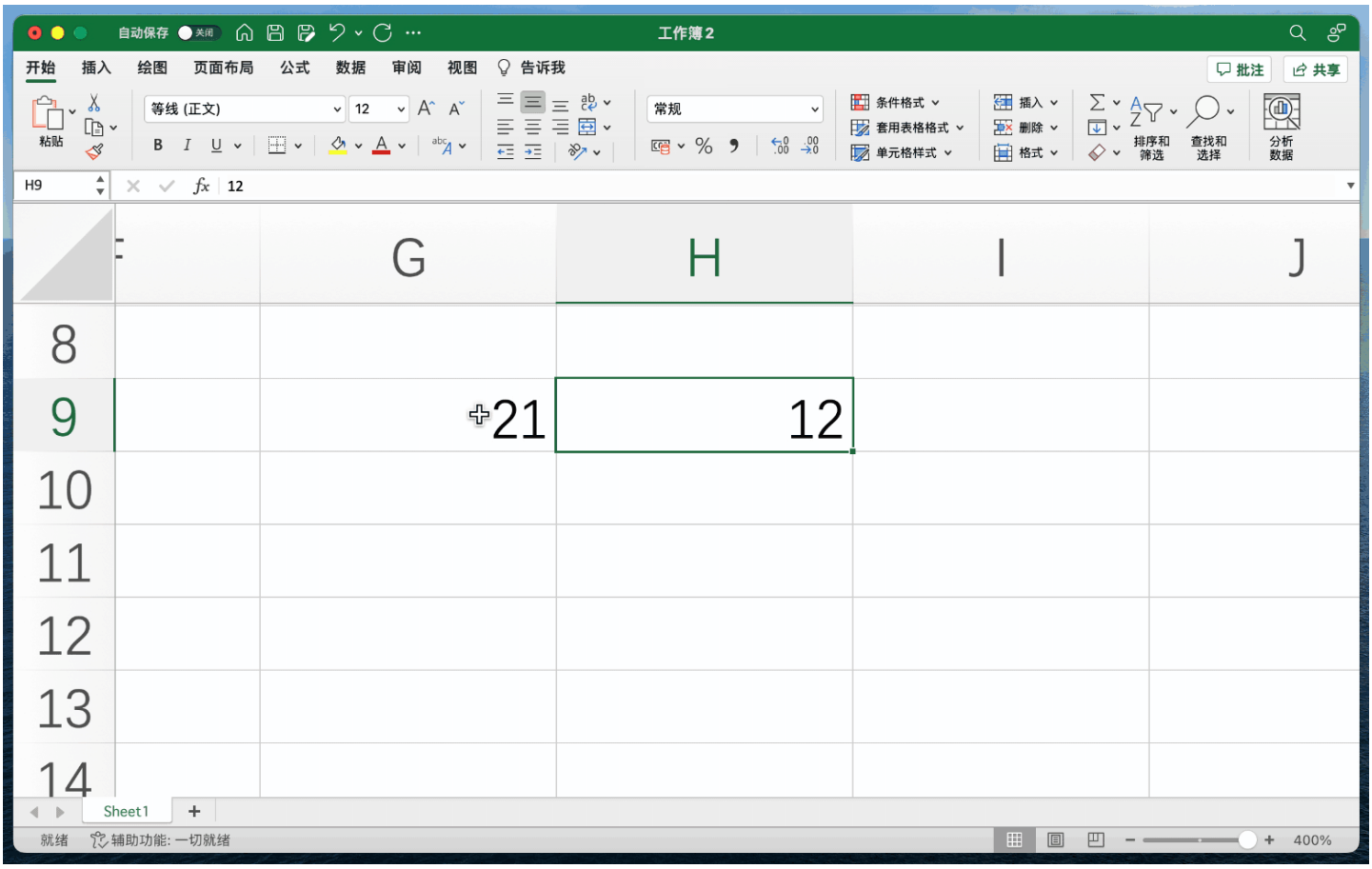


虽然 Notion Formula 与 Excel 的函数存在较大差异，但我们还是可以通过与 Excel 进行简单地对比，来更好地理解 Formula 的用法、特性，或者说限制。

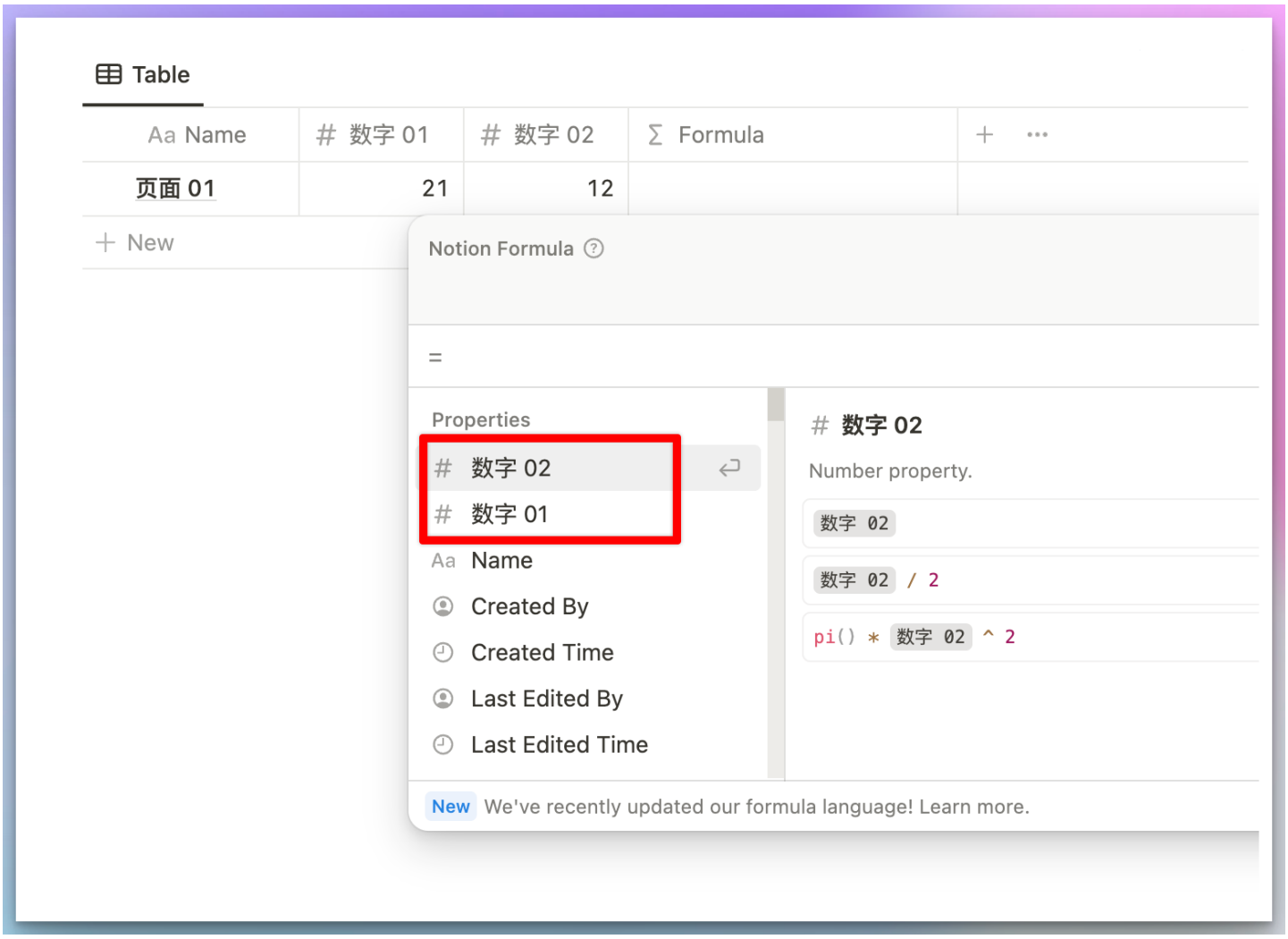
在 Excel 中，每一个单元格都可以用 `字母+数字` 的坐标来表示，所以当我们要计算下图中的

`21` 与 `12` 的和时，就可以在 **任意一个单元格** 内写上这么一个等式：

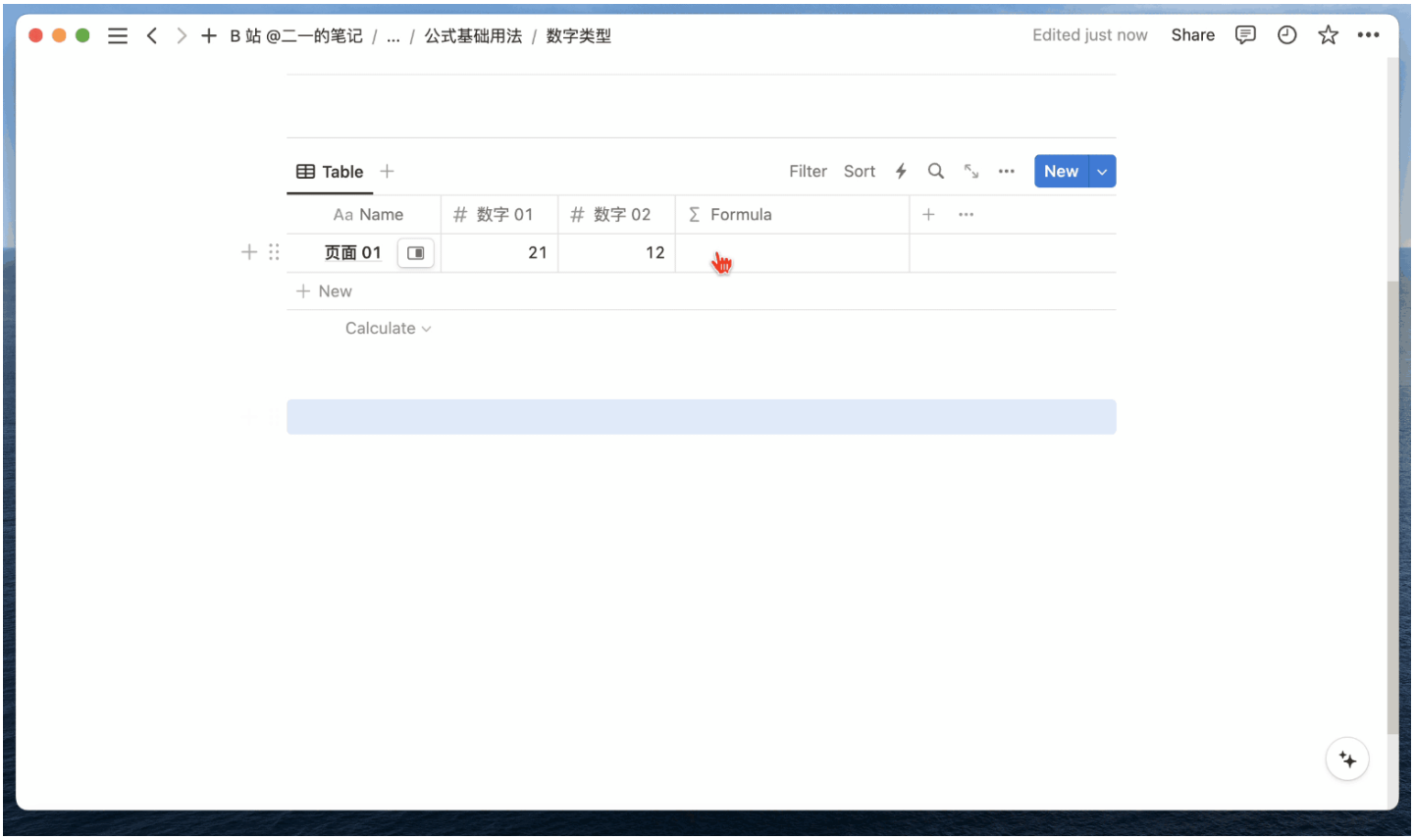
`=G9+H9`



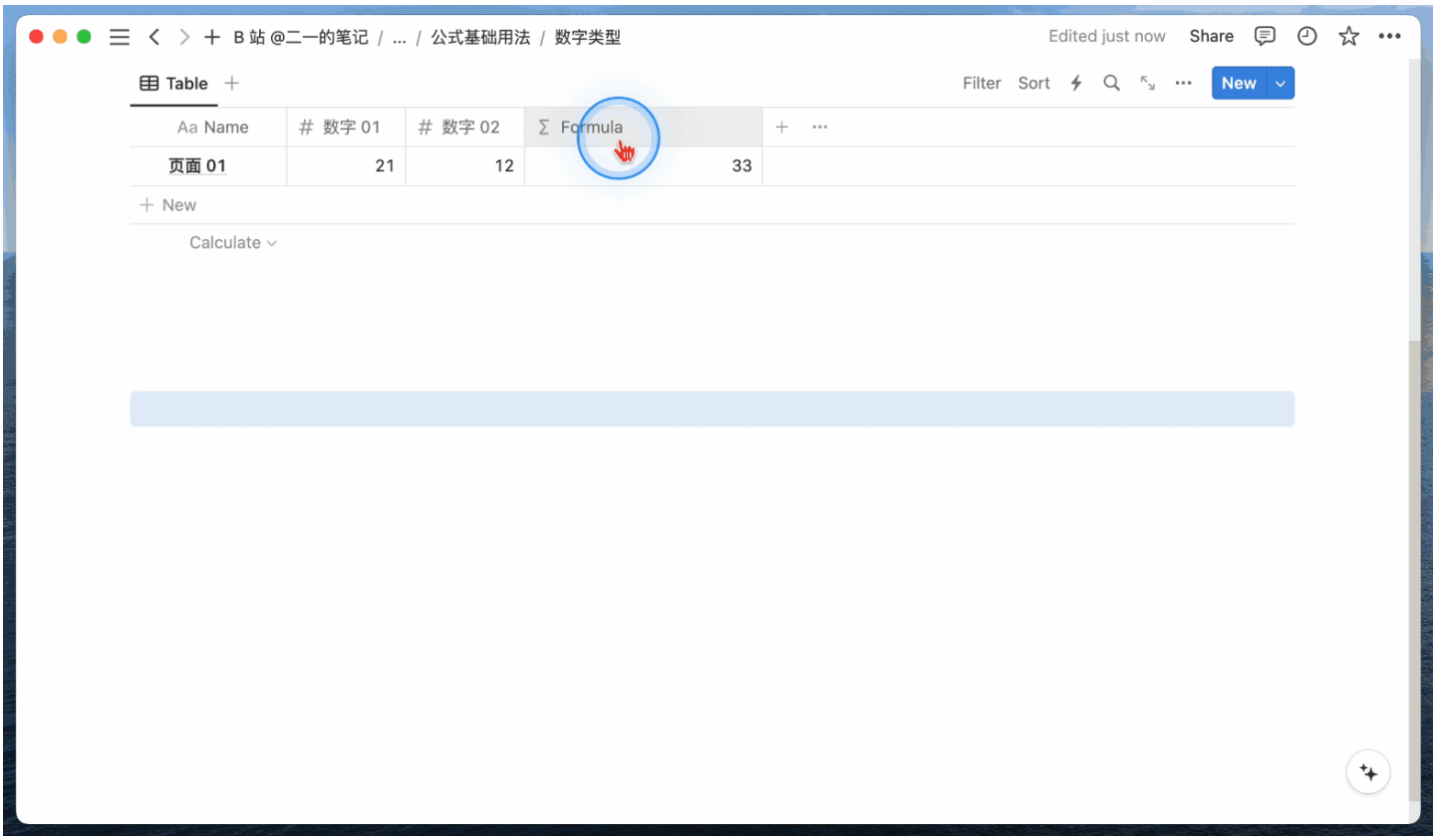
而在 Notion 中，如果要计算 `21` + `12` 的值，则需要先点击 Formula 字段，然后它就会弹出一个输入框，在输入框的左上角会显示当前数据库所有可用于计算的字段（Properties）：



接下来直接点击需要计算的字段，然后用数学运算符号 +
即可计算：

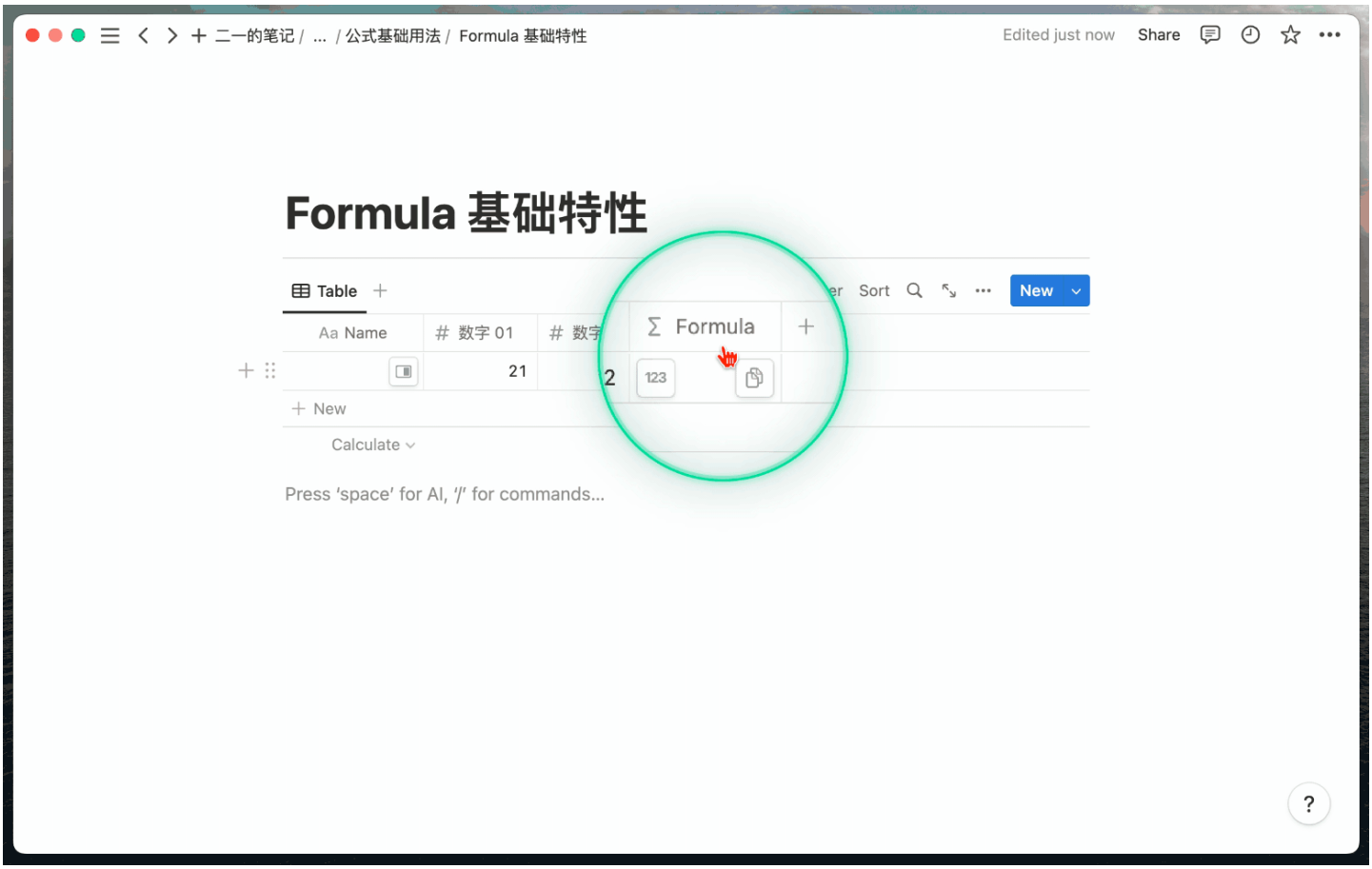


如果你想为 Formula
的计算结果添加货币单位、或者修改成百分比格式，可以编辑 Formula
字段，然后在 Number format 这一栏进行操作：



注意事项

Excel 可以为每个单元格设置不同的函数公式，但 Notion 的 Formula
会自动向下填充字段所在那一列的所有单元格，并且无法修改



Formula 只能计算本行（本页面）内的字段，无法像 Excel
那样跨行甚至跨表进行计算。例如在下图中，Notion 只能计算
`10`
与
`20`，但无法计算 `10` 与

`30`，因为这两个数字并不在同一个页面内：

Table

Aa Name	# 数字 01	# 数字 02	+ ···
页面 01	10	20	
页面 02		30	
+ New			

Formula 只能读取字段数据，不能改变字段的值，也不能创建新的字段

Formula 的四种数据类型

在前面的基础介绍中我们已经学会了基本的加减乘除计算方法，如下图所示

数字字段的计算

加法计算： X +Y `prop("X") + prop("Y")`

Table

Aa Name	# X	# Y	Σ f(x)	+ ···
	1	1	2	
	2	2	4	
+ New				
	Calculate	Calculate	Calculate	Calculate

利润计算：收入-支出 `prop("收入") - prop("支出")`

Table

Aa Name	# 收入	# 支出	Σ 盈利	+ ···
项目 01	100	200	-100	
项目 02	200	300	-100	
+ New				
			SUM -200	

总价计算：单价 * 数量 `prop("数量") * prop("单价")`

Table

Aa Name	# 数量	# 单价	Σ 总价	+ ···
商品 01	10	200000	$2,000,000.00	
商品 02	30	3000	$90,000.00	
+ New				

单价计算：总价 / 数量 `prop("总价") / prop("数量")`

Table

Aa Name	# 总价	# 数量	Σ 单价	+ ···
	1000	5	200	
	2000	10	200	
+ New				

但有些时候你可能会发现数据库中的某些「数字」无法直接计算，使用加号

`+`
得到的有可能不是两个数字之和，而是两个数字前后拼接的情况：

这是因为 Name 和 Tag
这两个字段都是字符串，而加号只能将字符串与字符串进行拼接。

加减乘除与四种数据类型

在 Notion
的公式计算中，我们可以将数据库的所有字段划分成四种类型，分别是数字、字符串、日期，以及布尔值：

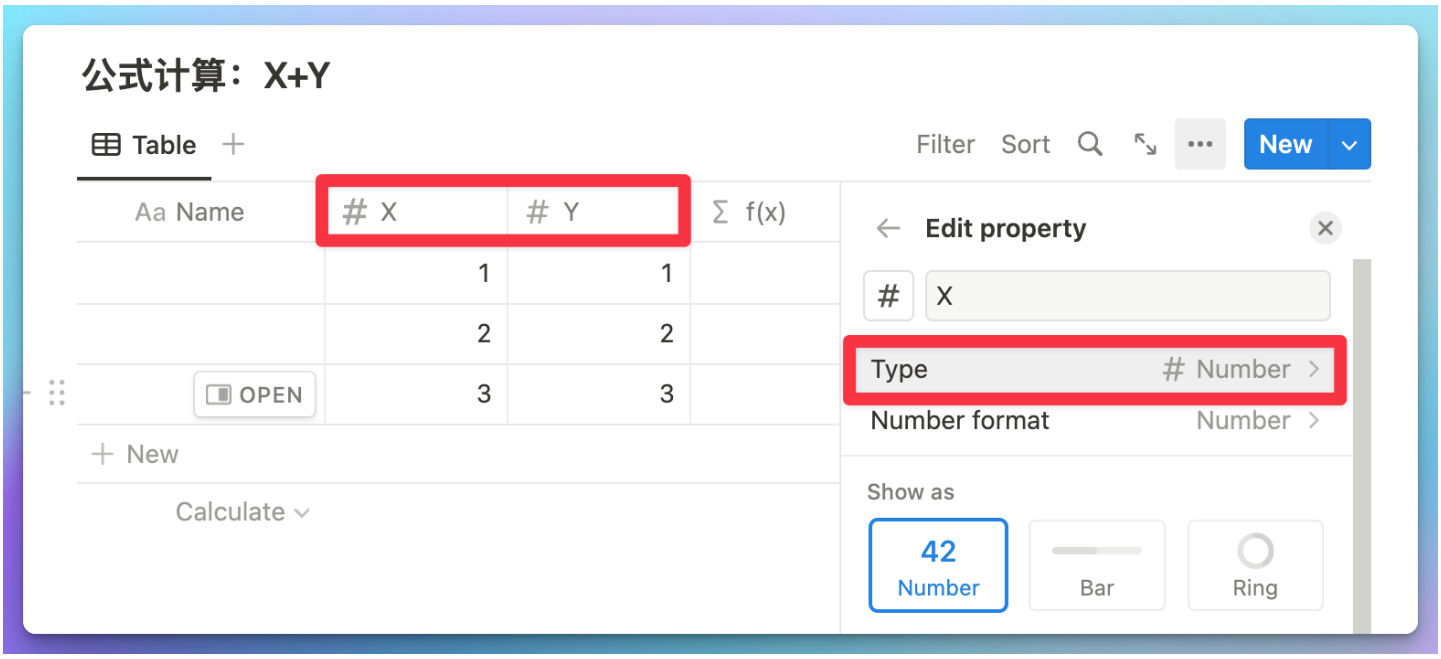
数据类型	字段（按默认顺序排序）
数字	Number
字符串	Title、Text、Select、Multi-Select、Person、File&Media、URL、Email、Phone、Created By、Last edited by
日期	Date、Created time、Last edited time
布尔值	Checkbox

想必新手看到这里可能就已经一脸懵了，先别担心，这个表格暂时不用记，之后遇到问题能够想起来有这么一回事就好。

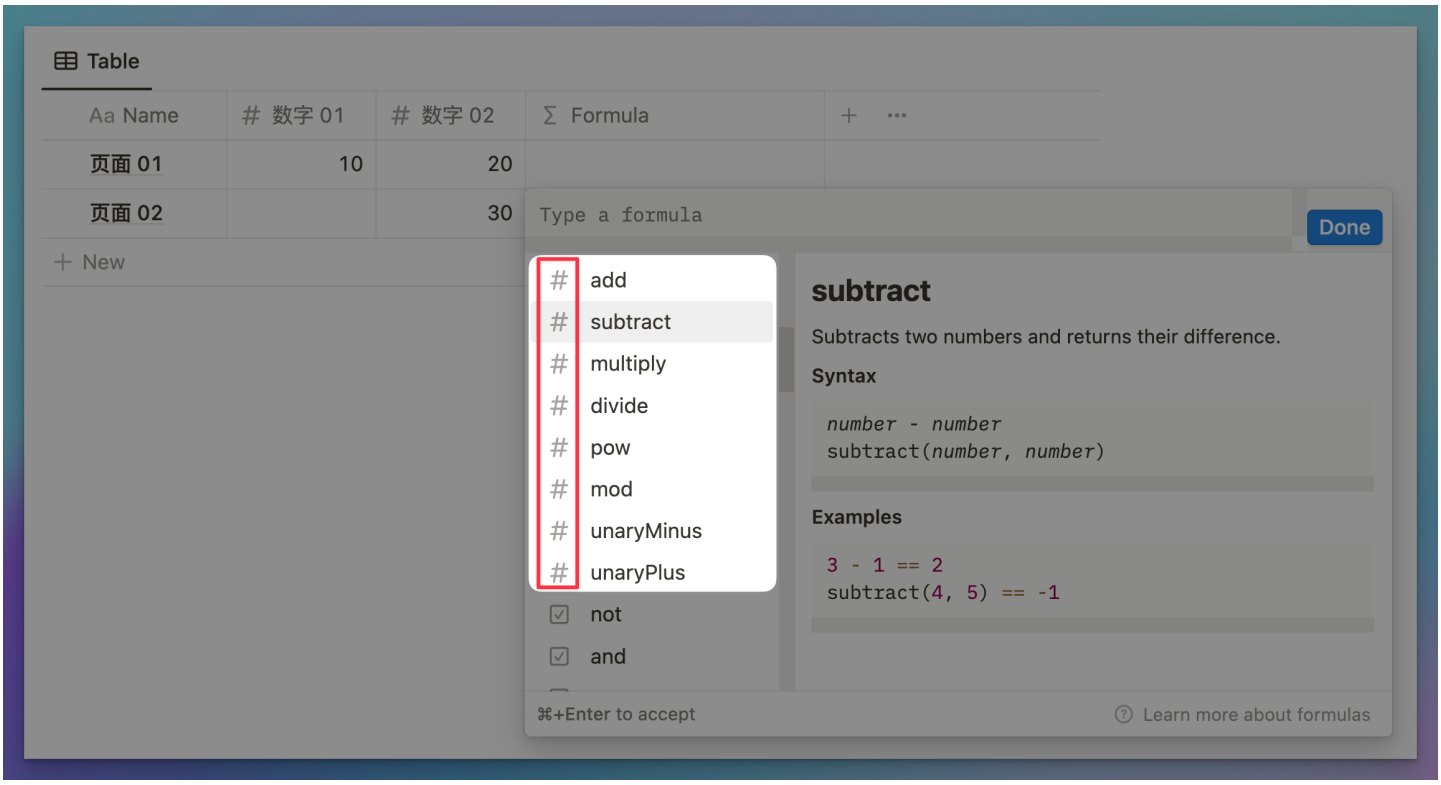
而在这四种数据类型中，我们先来讲讲最简单的「**数字**」，以及字段种类最多的「**字符串**」，后面的内容会循序渐进地带大家熟悉这四种数据类型各自的用法。

数字

在 Notion 的 Formula 中，只有同为数字字段（Number）
才能直接用四则运算符号进行加减乘除：



并且所有带有 `#` 标志的函数，计算得出的结果也都是数字，看到它就意味着我们可以直接对这些函数的计算结果继续进行四则计算：

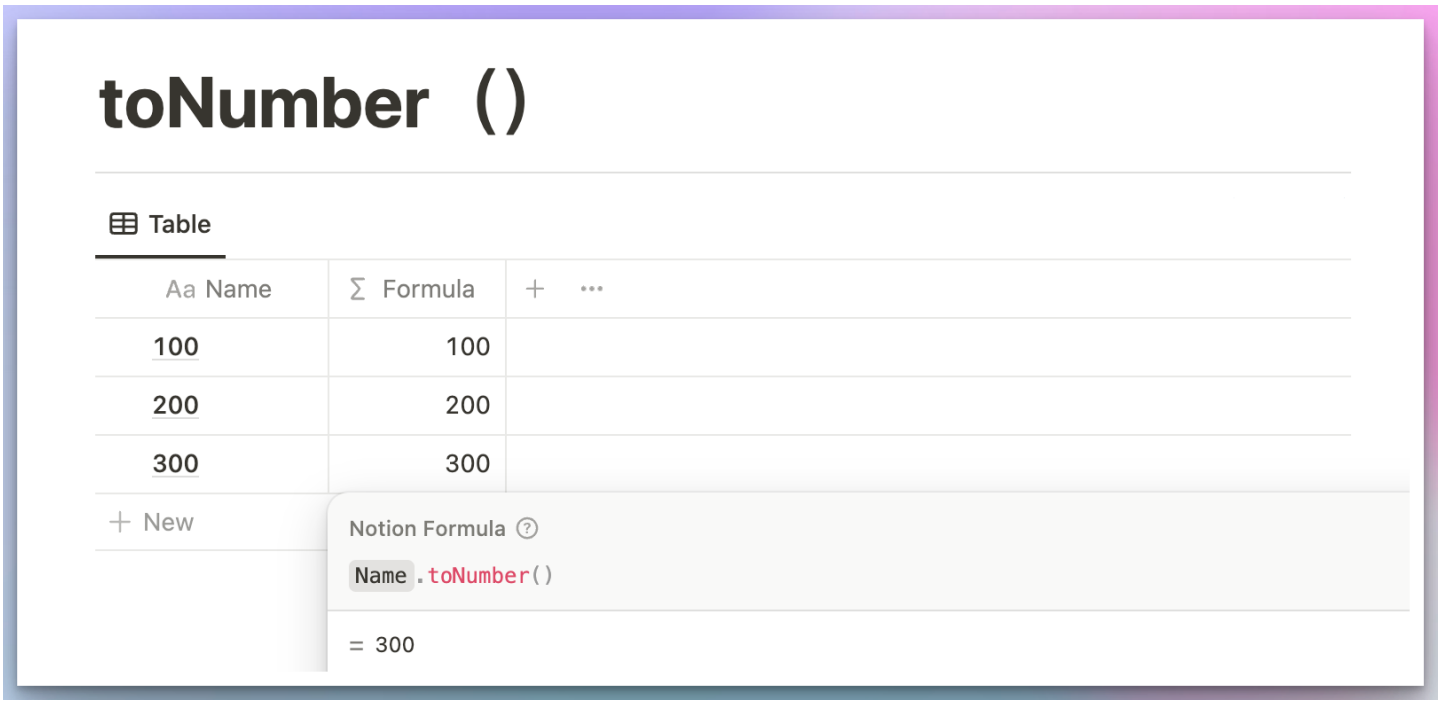


字符串类型

一般来说，除了日期和数字之外，你在数据库中使用的基本上都会是字符串类型的字段，例如 Select、Multi-select、Title 或者 Text，所以如果你需要计算的数字是写在这些字符串类型的字段里的话，则需要使用

`toNumber()` 函数，先将字符串转换为数字，或者干脆直接将 Text 字段修改为 Number 字段。

例如下图，Name 这一列是标题字段（Titile），所以我可以在 Formula 的公式书写区，使用 `prop("Name").toNumber()` 将标题转换成可以直接计算的数字：



将字符串转换为数字后，就可以直接进行四则运算了：

然后我们就可以通过选择不同的 Select 标签，来获得不同的计算结果：

当然我们也可以对**日期**或者**勾选框**进行计算，不过这部分的内容相对独立，所以我们会放在后面再做介绍。

更多运算符号

前面我们介绍了可以直接用四则运算符号来对数字进行计算，现在你还可以用下列这些函数来执行更多类型的操作：

运算符	作用	写法	结果
add	将两个数字相加并返回其总和，或者将两个字符串拼接起来	`add(1, 3)`	4
subtract	将两个数字相减并返回其差值	`subtract(4, 5)`	-1
multiply	将两个数字相乘并返回其乘积	`multiply(2, 10)`	20
divide	将两个数字相除并返回其商	`divide(12, 3)`	4
pow	计算指数次幂	`pow(2, 3)`	8
mod	将两个数字相除并返回其余数。	`mod(3, 3)`	0
unaryMinus	数字的负数	`unaryMinus(1)`	-1
unaryPlus	将其参数转换为数字	`unaryPlus(true)`	1

字符串拼接

如果需要拼接的数据都是字符串类型，那么只需要用加号 `+`
将它们写在一起即可：

Table

Aa Name	Text 1	Text 2	Text 3	Select	文本拼接 1
	Hello	World		!	HelloWorld!
	How	are	you	?	Howareyou?
	I	am	fine	.	Iamfine.
+ New					

Notion Formula

`Text 1` + `Text 2` + `Text 3` + `Select`

= Iamfine.

接下来你可能会发现拼接的文字之间缺少了空格，这该怎么办呢？很简单，只要在各个字段之间**手动添加一个英文引号下的空格**即可，就像这样

`" "`

Table

Aa Name	Text 1	Text 2	Text 3	Select	文本拼接结果
	Hello	World		!	Hello World !
	How	are	you	?	How are you?
	I	am	fine	.	I am fine.
+ New					

```
prop("Text 1") + " " + prop("Text 2")+ " " +prop("Text 3") +
prop("Select")
```

Done

数字、日期与字符串拼接

在下面这个案例中，我们要将文本、日期以及数字这三种属性不同的字段拼接在一起，然后生成一个最终的订单编号：

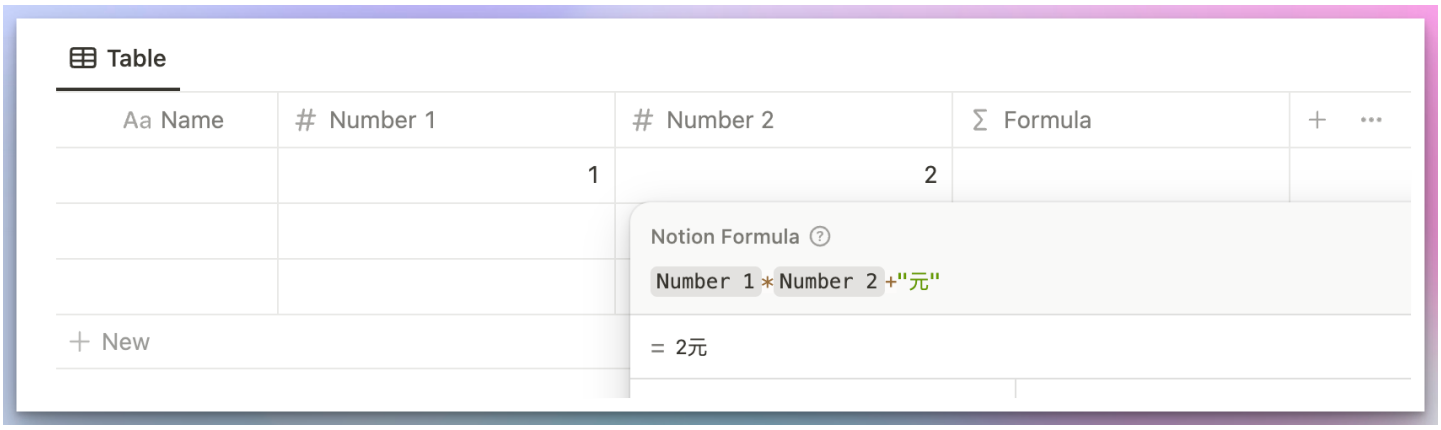
生成订单

Table

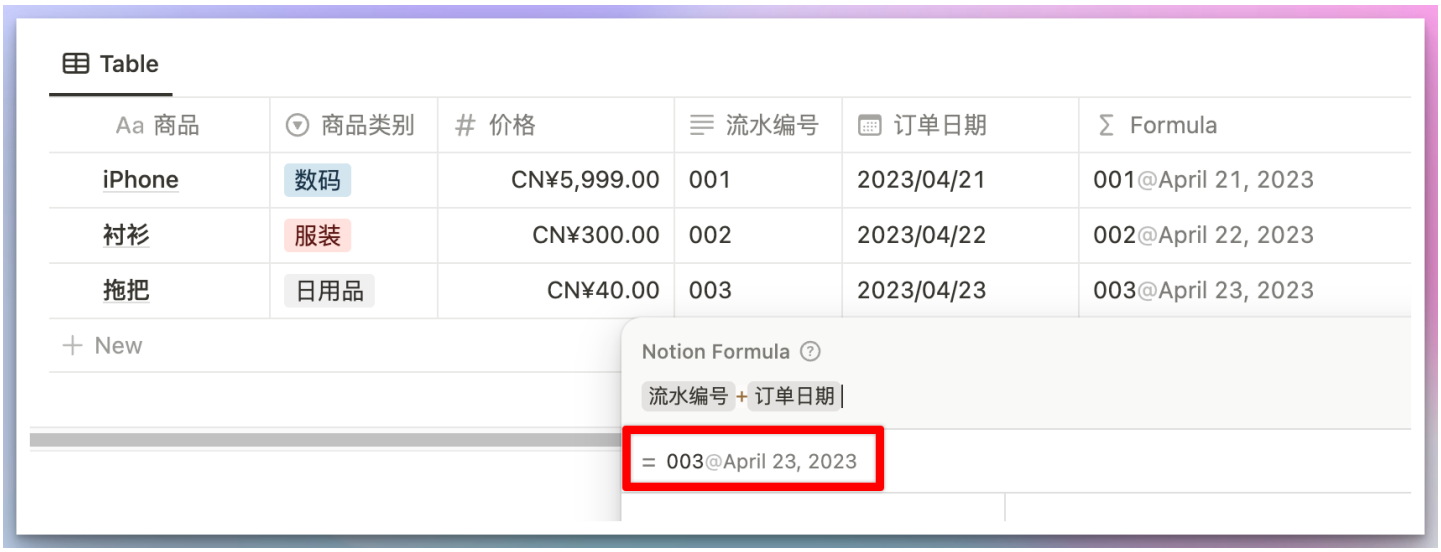
Name Select Number Text Date Formula

Aa 商品	商品类别	# 价格	流水编号	订单日期	生成订单
iPhone	数码	CN¥5,999.00	001	2023/04/21	001数码-iPhone599920230421
衬衫	服装	CN¥300.00	002	2023/04/22	002服装-衬衫30020230422
拖把	日用品	CN¥40.00	003	2023/04/23	003日用品-拖把4020230423
+ New					

在 Formula 1.0 中，数字与字符串是无法直接通过加号 `+`
来进行拼接的，但是更新了 Formula 2.0 之后，这种写法已经不会报错了：

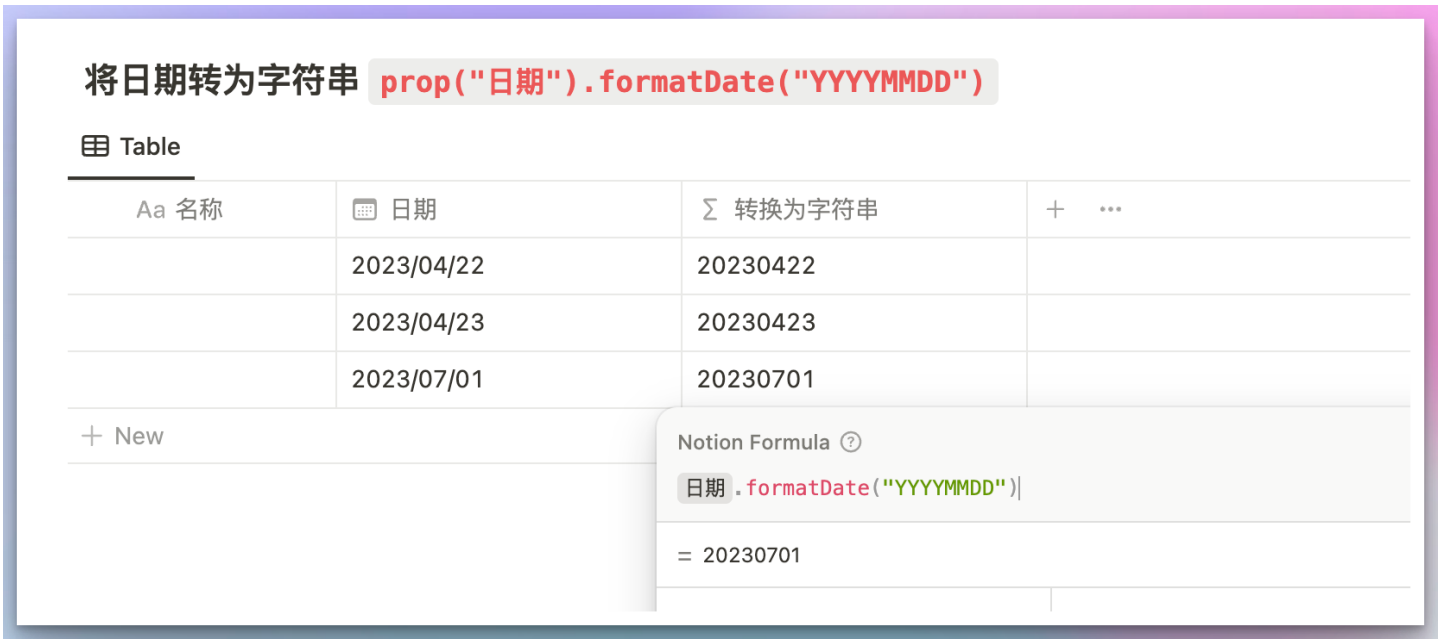


在 Formula 1.0 中，日期与字符串也无法直接用加号拼接，但是在 Formula 2.0 中，字符串和日期可以直接用加号拼接，并得到下图这样的结果：



但如果你希望生成的不是 `@April 23,2023` 这样的英文日期格式，而是 `20230423` 这样更符合我们阅读习惯的日期格式，就需要先使用 `formatDate()` 函数将日期进行格式化转换。

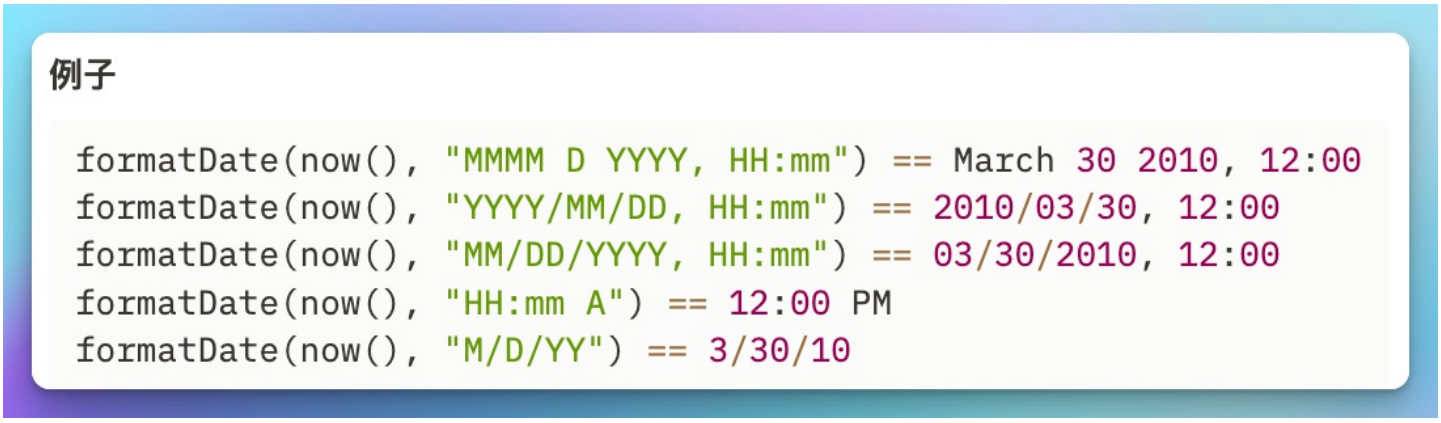
请特别注意，下图公式中 `formatDate` 的大写 D 不可写成小写的 d



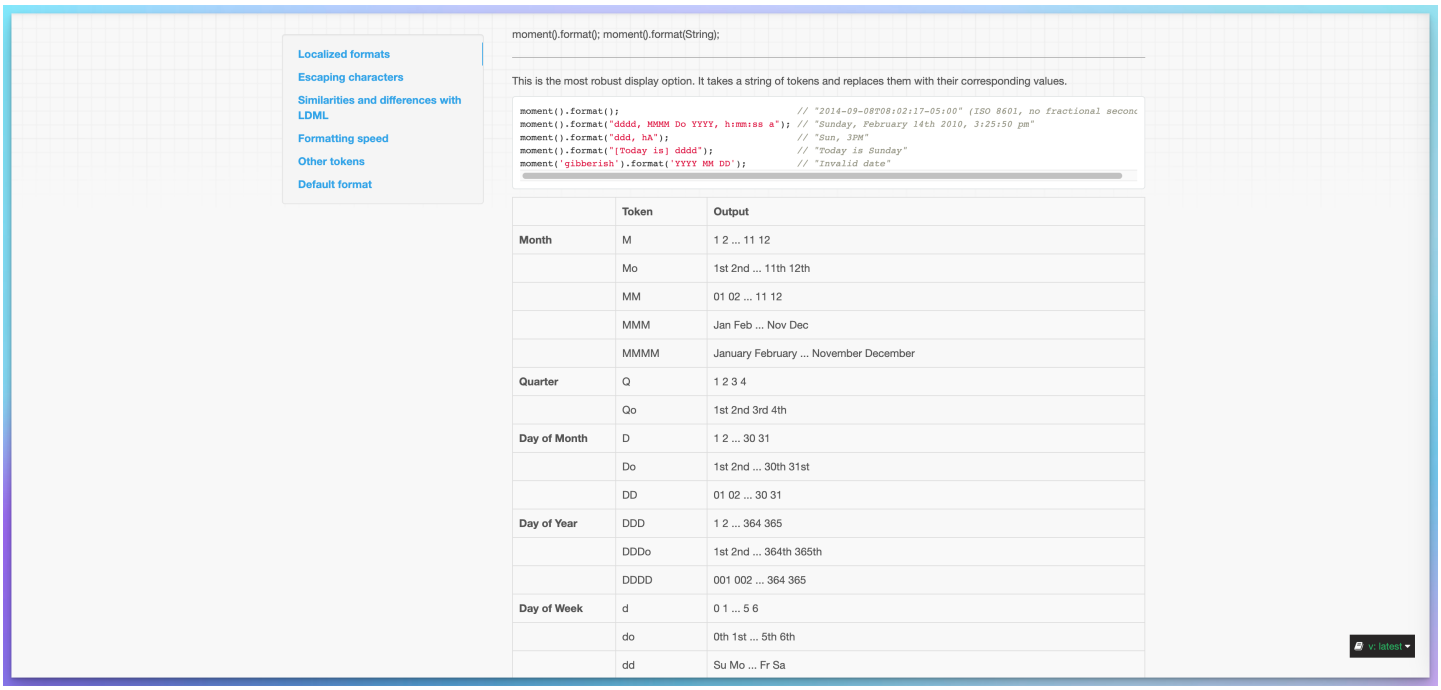
这里的 `YYYYMMDD` 是一个 8 位数的日期格式，分别代表年（year）、月（month）、日（day）。我们以 2023/04/01 这个日期为例，当我们调整 `YYYYMMDD` 的格式时会产生如下变化：

- `YYYYMMDD` 会将日期转化为 20230401
- `DDMMYYYY` 会将日期转化为 01042023
- `YYMMDD` 去掉了年份的前两位，会将日期转化为 230401
- `YYMDD` 去掉了月份的前一位，会将日期转化为 23401
- `YYMD` 去掉了天数的前一位，会日期转化为 2341

你可以随意地更改 Y、M、D 字母的个数以及排列顺序，然后通过 Notion 的动态变化来学习它们的更多用法。你也可以直接输入 `formatDate` 这个函数，然后从它底下的说明中观察和学习更多的参考案例：



或者你也可以一步到位，直接参考学习 Moment.js 的完整写法。



Localized formats
Escaping characters
Similarities and differences with LDML
Formatting speed
Other tokens
Default format

moment().format(); moment().format(String);

This is the most robust display option. It takes a string of tokens and replaces them with their corresponding values.

```
moment().format();                                  // "2014-09-08T08:02:17-05:00" (ISO 8601, no fractional second
moment().format("dddd, MMMM Do YYYY, h:mm:ss a"); // "Sunday, February 14th 2010, 3:25:50 pm"
moment().format("ddd, hA");                         // "Sun, 3PM"
moment().format("[Today is] dddd");                 // "Today is Sunday"
moment('gibberish').format('YYYY MM DD');           // "Invalid date"
```

	Token	Output
Month	M	1 2 ... 11 12
	Mo	1st 2nd ... 11th 12th
	MM	01 02 ... 11 12
	MMM	Jan Feb ... Nov Dec
	MMMM	January February ... November December
Quarter	Q	1 2 3 4
	Qo	1st 2nd 3rd 4th
Day of Month	D	1 2 ... 30 31
	Do	1st 2nd ... 30th 31st
	DD	01 02 ... 30 31
Day of Year	DDD	1 2 ... 364 365
	DDDo	1st 2nd ... 364th 365th
	DDDD	001 002 ... 364 365
Day of Week	d	0 1 ... 5 6
	do	0th 1st ... 5th 6th
	dd	Su Mo ... Fr Sa

v: latest

现在我们已经学会了将数字和日期转换成字符串的方法，所以接下来可以尝试将它们都拼接在一起了

生成订单

```
prop("流水编号") + prop("商品类别") + "-" + prop("商品") + prop("价格") + prop("订单日期").formatDate("YYYYMMDD")
```

Table

Aa 商品	商品类别	# 价格	流水编号	订单日期	Σ 生成订单
iPhone	数码	CN¥5,999.00	001	2023/04/21	001数码-iPhone599920230421
衬衫	服装	CN¥300			
拖把	日用品	CN¥40			

+ New

Notion Formula

流水编号 + 商品类别 + "-" + 商品 + 价格 + 订单日期.formatDate("YYYYMMDD")

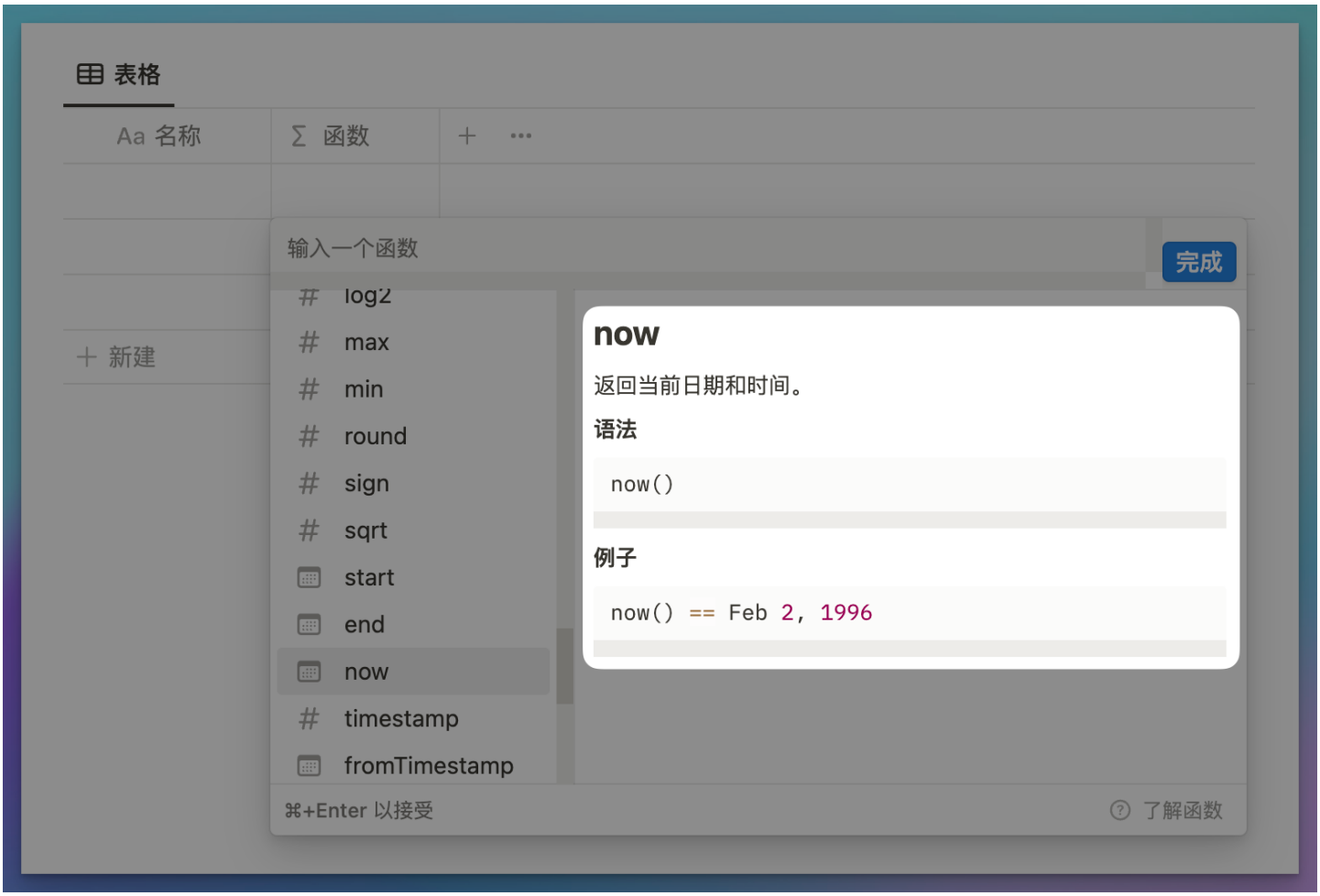
= 003日用品-拖把4020230423

公式如下：

```
prop("流水编号") + prop("商品类别") + "-" + prop("商品") + prop("价格") + prop("订单日期").formatDate("YYYYMMDD"
)
```

Formula 易错点

1. 所有函数的大小写敏感，如果你发现有些公式好像写对了，但却无法生效，则可以考虑检查一下该大写的是否大写了
2. 所有函数的括号、逗号、引号都需要在英文输入法下使用
3. 所有公式的括号数量都需要书写正确
4. 对新手来说，学习 Formula 的最大问题可能还是在于 Notion
   自带的公式说明案例看不懂，如果是这样的话，我会建议各位在浏览器上装一个汉化插件（安装方法见《Notion 新手最常见的 6 个问题》），汉化之后至少能看得懂每个函数的基础用途都是什么了：



> 本文由简悦 SimpRead 转码