

EXCEL 函数手册

SUNLION | 2009

目录

一. 函数应用基础	1
1. 函数和公式	1
[1]. 什么是函数	1
[2]. 什么是公式	1
2. 函数参数	1
[1]. 常量	1
[2]. 逻辑值	1
[3]. 数组	2
[4]. 错误值	2
[5]. 单元格引用	2
[6]. 嵌套函数	3
[7]. 名称和标志	3
3. 函数输入方法	3
[1]. “插入函数”对话框	3
[2]. 编辑栏输入	4
二. 函数速查一览	4
1. 数据库函数	4
[1]. DAVERAGE	4
[2]. DCOUNT	4
[3]. DCOUNTA	4
[4]. DGET	5
[5]. DMAX	5
[6]. DMIN	5
[7]. DPRODUCT	5
[8]. DSTDEV	5
[9]. DSTDEVP)	5
[10]. DSUM	6
[11]. DVAR	6
[12]. DVARP	6
[13]. GETPIVOTDATA	6
2. 日期与时间函数	6
[1]. DATE	6
[2]. DATEVALUE	7
[3]. DAY	7
[4]. DAYS360	7
[5]. EDATE	7
[6]. EOMONTH	8
[7]. HOUR	8
[8]. MINUTE	8
[9]. MONTH	8
[10]. NETWORKDAYS	9
[11]. NOW	9

[12].	SECOND	9
[13].	TIME	9
[14].	TIMEVALUE	9
[15].	TODAY	10
[16].	WEEKDAY	10
[17].	WEEKNUM	10
[18].	WORKDAY	10
[19].	YEAR	11
[20].	YEARFRAC	11
3. 外部函数		11
[21].	EUROCONVERT	11
[22].	SQL.REQUEST	11
4. 工程函数		12
[1].	BESSELI	12
[2].	BESSELJ	12
[3].	BESSELK	12
[4].	BESSELY	12
[5].	BIN2DEC	12
[6].	BIN2HEX	12
[7].	BIN2OCT	13
[8].	COMPLEX	13
[9].	CONVERT	13
[10].	DEC2BIN	13
[11].	DEC2HEX	13
[12].	DEC2OCT	13
[13].	DELTA	13
[14].	ERF	14
[15].	ERFC	14
[16].	GESTEP	14
[17].	HEX2BIN	14
[18].	HEX2DEC	14
[19].	HEX2OCT	14
[20].	IMABS	14
[21].	IMAGINARY	15
[22].	IMARGUMENT	15
[23].	MCONJUGATE	15
[24].	IMCOS	15
[25].	IMDIV	15
[26].	IMEXP	15
[27].	IMLN	15
[28].	IMLOG10	15
[29].	IMLOG2	16
[30].	IMPOWER	16
[31].	IMPRODUCT	16

[32].	IMREAL	16
[33].	IMSIN.....	16
[34].	IMSQRT	16
[35].	IMSUB.....	16
[36].	IMSUM	17
[37].	OCT2BIN.....	17
[38].	OCT2DEC	17
[39].	OCT2HEX.....	17
5. 财务函数.....		17
[1].	ACCRINT.....	17
[2].	ACCRINTM	17
[3].	AMORDEGRC	18
[4].	AMORLINC	18
[5].	COUPDAYBS	18
[6].	COUPDAYS	18
[7].	COUPDAYSNC.....	18
[8].	COUPNUM	19
[9].	COUPPCD.....	19
[10].	CUMIPMT.....	19
[11].	CUMPRINC.....	19
[12].	DB	19
[13].	DDB.....	19
[14].	DISC.....	20
[15].	DOLLARDE	20
[16].	DOLLARFR	20
[17].	DURATION	20
[18].	EFFECT	20
[19].	FV	20
[20].	FVSCHEDULE.....	21
[21].	INTRATE.....	21
[22].	IPMT	21
[23].	IRR	21
[24].	ISPMT	21
[25].	MDURATION.....	21
[26].	MIRR.....	22
[27].	NOMINAL	22
[28].	NPER	22
[29].	NPV	22
[30].	ODDFPRICE.....	22
[31].	ODDFYIELD	23
[32].	ODDLPRICE.....	23
[33].	ODDLYIELD.....	23
[34].	PMT	23
[35].	PPMT	24

[36].	PRICE	24
[37].	PRICEDISC	24
[38].	PRICEMAT	24
[39].	PV	24
[40].	RATE.....	25
[41].	RECEIVED	25
[42].	SLN	25
[43].	SYD	25
[44].	TBILLEQ	25
[45].	TBILLPRICE	25
[46].	TBILLYIELD	26
[47].	VDB.....	26
[48].	XIRR.....	26
[49].	XNPV.....	26
[50].	YIELD.....	26
[51].	YIELDDISC	26
[52].	YIELDMAT	27
6.	信息函数.....	27
[1].	CELL	27
[2].	ERROR.TYPE.....	27
[3].	INFO	27
[4].	IS 类函数.....	28
[5].	ISEVEN.....	28
[6].	ISODD	28
[7].	N.....	29
[8].	NA	29
[9].	TYPE.....	29
7.	逻辑运算符	29
[1].	AND	29
[2].	FALSE.....	30
[3].	IF.....	30
[4].	NOT	30
[5].	OR	30
[6].	TRUE	31
8.	查找和引用函数.....	31
[1].	ADDRESS.....	31
[2].	AREAS	31
[3].	CHOOSE.....	31
[4].	COLUMN	32
[5].	COLUMNS	32
[6].	HLOOKUP	32
[7].	HYPERLINK	32
[8].	INDEX	33
[9].	INDIRECT.....	33

[10].	LOOKUP	33
[11].	MATCH	34
[12].	OFFSET	34
[13].	ROW	34
[14].	ROWS	35
[15].	RTD.....	35
[16].	TRANSPOSE	35
[17].	VLOOKUP	35
9. 数学和三角函数.....		36
[18].	ABS.....	36
[19].	ACOS	36
[20].	ACOSH	36
[21].	ASIN	36
[22].	ASINH.....	36
[23].	ATAN	36
[24].	ATAN2	37
[25].	ATANH	37
[26].	CEILING.....	37
[27].	COMBIN	37
[28].	COS	37
[29].	COSH.....	38
[30].	COUNTIF.....	38
[31].	DEGREES	38
[32].	EVEN	38
[33].	EXP	38
[34].	FACT	38
[35].	FACTDOUBLE	39
[36].	FLOOR	39
[37].	GCD	39
[38].	INT	39
[39].	LCM	39
[40].	LN.....	40
[41].	LOG	40
[42].	LOG10	40
[43].	MDETERM	40
[44].	MINVERSE.....	40
[45].	MMULT	40
[46].	MOD	41
[47].	MROUND.....	41
[48].	MULTINOMIAL	41
[49].	ODD.....	41
[50].	PI	41
[51].	POWER	42
[52].	PRODUCT	42

[53].	QUOTIENT	42
[54].	RADIANS	42
[55].	RAND	42
[56].	RANDBETWEEN	43
[57].	ROMAN	43
[58].	ROUND	43
[59].	ROUNDDOWN	43
[60].	ROUNDUP	43
[61].	SERIESSUM	44
[62].	SIGN	44
[63].	SIN	44
[64].	SINH	44
[65].	SQRT	44
[66].	SQRTPI	45
[67].	SUBTOTAL	45
[68].	SUM	45
[69].	SUMIF	45
[70].	SUMPRODUCT	46
[71].	SUMSQ	46
[72].	SUMX2MY2	46
[73].	SUMX2PY2	46
[74].	SUMXMY2	46
[75].	TAN	46
[76].	TANH	47
[77].	TRUNC	47
10.	统计函数	47
[78].	AVEDEV	47
[79].	AVERAGE	47
[80].	AVERAGEA	47
[81].	BETADIST	48
[82].	BETAINV	48
[83].	BINOMDIST	48
[84].	CHIDIST	48
[85].	CHIINV	49
[86].	CHITEST	49
[87].	CONFIDENCE	49
[88].	CORREL	49
[89].	COUNT	50
[90].	COUNTA	50
[91].	COUNTBLANK	50
[92].	COUNTIF	50
[93].	COVAR	50
[94].	CRITBINOM	51
[95].	DEVSQ	51

[96].	EXPONDIST	51
[97].	FDIST	51
[98].	FINV	51
[99].	FISHER	52
[100].	FISHERINV	52
[101].	FORECAST	52
[102].	FREQUENCY	52
[103].	FTEST.....	52
[104].	GAMMADIST	53
[105].	GAMMAINV	53
[106].	GAMMALN	53
[107].	GEOMEAN	53
[108].	GROWTH	53
[109].	HARMEAN	54
[110].	HYPGEOMDIST.....	54
[111].	INTERCEPT	54
[112].	KURT.....	54
[113].	LARGE	55
[114].	LINEST.....	55
[115].	LOGEST.....	55
[116].	LOGINV	55
[117].	LOGNORMDIST.....	56
[118].	MAX.....	56
[119].	MAXA	56
[120].	MEDIAN	56
[121].	MIN	56
[122].	MINA.....	57
[123].	MODE	57
[124].	NEGBINOMDIST	57
[125].	NORMDIST.....	57
[126].	NORMSINV	57
[127].	NORMSDIST	58
[128].	NORMSINV	58
[129].	PEARSON	58
[130].	PERCENTILE	58
[131].	PERCENTRANK.....	58
[132].	PERMUT.....	59
[133].	POISSON	59
[134].	PROB.....	59
[135].	QUARTILE.....	59
[136].	RANK	59
[137].	RSQ	60
[138].	SKEW	60
[139].	SLOPE	60

[140].	SMALL	60
[141].	STANDARDIZE.....	61
[142].	STDEV.....	61
[143].	STDEVA.....	61
[144].	STDEVP	61
[145].	STDEVPA.....	62
[146].	STEYX	62
[147].	TDIST.....	62
[148].	TINV.....	62
[149].	TREND	62
[150].	TRIMMEAN.....	63
[151].	TTEST.....	63
[152].	VAR.....	63
[153].	VARA	63
[154].	VARP.....	64
[155].	VARPA.....	64
[156].	WEIBULL	64
[157].	ZTEST.....	64
11.	文本和数据函数.....	64
[158].	ASC.....	64
[159].	CHAR.....	65
[160].	CLEAN	65
[161].	CODE	65
[162].	CONCATENATE	65
[163].	DOLLAR 或 RMB.....	65
[164].	EXACT.....	66
[165].	FIND	66
[166].	FINDB	66
[167].	FIXED.....	66
[168].	JIS.....	67
[169].	LEFT 或 LEFTB.....	67
[170].	LEN 或 LENB	67
[171].	LOWER.....	67
[172].	MID 或 MIDB.....	67
[173].	PHONETIC	68
[174].	PROPER	68
[175].	REPLACE 或 REPLACEB	68
[176].	REPT	68
[177].	RIGHT 或 RIGHTB	68
[178].	SEARCH 或 SEARCHB	69
[179].	SUBSTITUTE	69
[180].	T.....	69
[181].	TEXT	69
[182].	TRIM	70

[183]. UPPER	70
[184]. VALUE	70
[185]. WIDECCHAR	70
三. 函数应用案例——算账理财	71
1. 零存整取储蓄	71
[1]. 函数分解	71
[2]. 实例分析	71
2. 还贷金额	71
[1]. 函数分解	71
[2]. 实例分析	71
3. 保险收益	72
[1]. 函数分解	72
[2]. 实例分析	72
4. 个税缴纳金额	72
四. 函数应用案例——信息统计	73
1. 性别输入	73
[1]. 函数分解	73
[2]. 实例分析	73
2. 出生日期输入	74
[1]. 函数分解	74
[2]. 实例分析	74
3. 职工信息查询	74
[1]. 函数分解	74
[2]. 实例分析	75
4. 职工性别统计	76
[1]. 函数分解	76
[2]. 实例分析	76
5. 年龄统计	76
[1]. 函数分解	76
[2]. 实例分析	76
6. 名次值统计	77
[1]. 函数分解	77
[2]. 实例分析	77
7. 位次阈值统计	77
[1]. 函数分解	77
[2]. 实例分析	77
五. 函数应用案例——管理计算	78
1. 授课日数	78
[1]. 函数分解	78
[2]. 实例分析	78
2. 折旧值计算	78
[1]. 函数分解	78
[2]. 实例分析	79
3. 客流均衡度计算	79

[1]. 函数分解	79
[2]. 实例分析	79
4. 销售额预测	79
[3]. 函数分解	79
[1]. 实例分析	80
5. 客流与营业额的相关分析	80
[1]. 函数分解	80
[2]. 实例分析	80

一. 函数应用基础

1. 函数和公式

[1]. 什么是函数

Excel 函数即是预先定义，执行计算、分析等处理数据任务的特殊公式。以常用的求和函数 SUM 为例，它语法是“SUM(number1,number2,.....)”。其中“SUM”称为函数名称，一个函数只有唯一的一个名称，它决定了函数的功能和用途。函数名称后紧跟左括号，接着是用逗号分隔的称参数的内容，最后用一个右括号表示函数结束。

参数是函数中最复杂的组成部分，它规定了函数的运算对象、顺序或结构等。使得用户可以对某个单元格或区域进行处理，如分析存款利息、确定成绩名次、计算三角函数值等。按照函数的来源，Excel 函数可以分为内置函数和扩展函数两大类。前者只要启动了 Excel，用户就可以使用它们；而后者必须通过单击“工具→加载宏”菜单命令加载，然后才能像内置函数那样使用。

[2]. 什么是公式

函数与公式既有区别又互相联系。如果说前者是 Excel 预先定义好的特殊公式，后者就是由用户自行设计对工作表进行计算和处理的公式。以公式“=SUM(E1:H1)*A1+26”为例，它要以等号“=”开始，其内部可以包括函数、引用、运算符和常量。上式中的“SUM(E1:H1)”是函数，“A1”则是对单元格 A1 的引用(使用其中存储的数据)，“26”则是常量，“*”和“+”则是算术运算符(另外还有比较运算符、文本运算符和引用运算符)。

如果函数要以公式的形式出现，它必须有两个组成部分，一个是函数名称前面的等号，另一个则是函数本身。

2. 函数参数

函数右边括号中的部分称参数，假如一个函数可以使用多个参数，那参数与参数之间使用半角逗号进行分隔。参数可以是常量(数字和文本)、逻辑值(例如 TRUE 或 FALSE)、数组、错误值(例如 #N/A)或单元格引用(例如 E1:H1)，甚至可以是另一个或几个函数等参数的类型和位置必须满足函语法的要求，否则将返回错误信息。

[1]. 常量

常量是直接输入到单元格或公式中的数字或文本，或由名称所代表的数字或文本值，例如数字“2890.56”、日期“2003-8-19”和文本“黎明”都是常量。但是公式或由公式计算出的结果都不是常量，因为只要公式的参数发生了变化，它自身或计算出来的结果就会发生变化。

[2]. 逻辑值

逻辑值是比较特殊的一类参数，它只有 TRUE(真)或 FALSE(假)两种类型。例如在公式

“=IF(A3=0, "", A2/A3)” 中, “A3=0” 就是一个可以返回 TRUE(真)或 FALSE(假)两种结果的参数。当 “A3=0” 为 TRUE(真)时在公式所在单元格中填入 “0”, 否则在单元格中填入 “A2/A3” 的计算结果。

[3]. 数组

数组用于可产生多个结果, 或可以对存放在行和列中的一组参数进行计算的公式。Excel 中有常量和区域两类数组。前者放在 “{}” (按下 Ctrl+Shift+Enter 组合键自动生成) 内部, 而且内部各列的数值要用逗号 “,” 隔开, 各行的数值要用分号 “;” 隔开。假如你要表示第 1 行中的 56、78、89 和第 2 行中的 90、76、80, 就应该建立一个 2 行 3 列的常量数组 “{56,78,89;90,76,80}”。区域数组是一个矩形的单元格区域, 该区域中的单元格共用一个公式。例如公式 “=TREND(B1:B3,A1:A3)” 作为数组公式使用时, 它所引用的矩形单元格区域 “B1:B3,A1:A3” 就是一个区域数组。

[4]. 错误值

使用错误值作为参数的主要是信息函数, 例如 “ERROR.TYPE” 函数就是以错误值作为参数。它语法为 “ERROR.TYPE(error_val)”, 如果其中的参数是 #NUM!, 则返回数值 “6”。

[5]. 单元格引用

单元格引用是函数中最常见的参数, 引用的目的在于标识工作表单元格或单元格区域, 并指明公式或函数所使用的数据的位置, 便于它们使用工作表各处的数据, 或者在多个函数中使用同一个单元格的数据。还可以引用同一工作簿不同工作表的单元格, 甚至引用其他工作簿中的数据。根据公式所在单元格的位置发生变化时, 单元格引用的变化情况, 我们可以引用分为相对引用、绝对引用和混合引用三种类型。

以存放在 F2 单元格中的公式 “=SUM(A2:E2)” 为例, 当公式由 F2 单元格复制到 F3 单元格以后, 公式中的引用也会变化为 “=SUM(A3:E3)”。若公式自 F 列向下继续复制, “行标” 每增加 1 行, 公式中的行标也自动加 1。如果上述公式改为 “=SUM(\$A\$3:\$E\$3)”, 则无论公式复制到何处, 其引用的位置始终是 “A3:E3” 区域。混合引用有 “绝对列和相对行”, 或是 “绝对行和相对列” 两种形式。前者如 “=SUM(\$A3:\$E3)”, 后者如 “=SUM(A\$3:E\$3)”。

上面的几个实例引用的都是同一工作表中的数据, 如果要分析同一工作簿中多张工作表上的数据, 就要使用三维引用。假如公式放在工作表 Sheet1 的 C6 单元格, 要引用工作表 Sheet2 的 “A1:A6” 和 Sheet3 的 “B2:B9” 区域进行求和运算, 则公式中的引用形式为 “=SUM(Sheet2!A1:A6,Sheet3!B2:B9)”。也就是说三维引用中不仅包含单元格或区域引用, 还要在前面加上带 “!” 的工作表名称。假如你要引用的数据来自另一个工作簿, 如工作簿 Book1 中的 SUM 函数要绝对引用工作簿 Book2 中的数据, 其公式为 “=SUM([Book2]Sheet1!\$A\$1:\$A\$8,[Book2]Sheet2!\$B\$1:\$B\$9)”, 也就是在原来单元格引用的前面加上 “[Book2]Sheet1!”。放在中括号里面的是工作簿名称, 带 “!” 的则是其中的工作表名称。即是跨工作簿引用单元格或区域时, 引用对象的前面必须用 “!” 作为工作表分隔符, 再用中括号作为工作簿分隔符。不过三维引用的要受到较多的限制, 例如不能使用数组公式等。提示: 上面介绍的是 Excel 默认的引用方式, 称为 “A1 引用样式”。如果你要计算处在 “宏” 内的行和列, 必须使用 “R1C1 引用样式”。在这种引用

样式中, Excel 使用“R”加“行标”和“C”加“列标”的方法指示单元格位置。启用或关闭 R1C1 引用样式必须单击“工具→选项”菜单命令, 打开对话框的“常规”选项卡, 选中或清除“设置”下的“R1C1 引用样式”选项。由于这种引用样式很少使用, 限于篇幅本文不做进一步介绍。

[6]. 嵌套函数

除了上面介绍的情况外, 函数也可以是嵌套的, 即一个函数是另一个函数的参数, 例如“=IF(OR(RIGHTB(E2,1)="1", RIGHTB(E2,1)="3", RIGHTB(E2,1)="5", RIGHTB(E2,1)="7", RIGHTB(E2,1)="9"),"男","女")”。其中公式中的 IF 函数使用了嵌套的 RIGHTB 函数, 并将后者返回的结果作为 IF 的逻辑判断依据。

[7]. 名称和标志

为了更加直观地标识单元格或单元格区域, 我们可以给它们赋予一个名称, 从而在公式或函数中直接引用。例如“B2:B46”区域存放着学生的物理成绩, 求解平均分的公式一般是“=AVERAGE(B2:B46)”。

在给 B2:B46 区域命名为“物理分数”以后, 该公式就可以变为“=AVERAGE(物理分数)”, 从而使公式变得更加直观。给一个单元格或区域命名的方法是: 选中要命名的单元格或单元格区域, 鼠标单击编辑栏顶端的“名称框”, 在其中输入名称后回车。也可以选中要命名的单元格或单元格区域, 单击“插入→名称→定义”菜单命令, 在打开的“定义名称”对话框中输入名称后确定即可。如果你要删除已经命名的区域, 可以按相同方法打开“定义名称”对话框, 选中你要删除的名称删除即可。由于 Excel 工作表多数带有“列标志”。例如一张成绩统计表的首行通常带有“序号”、“姓名”、“数学”、“物理”等“列标志”(也可以称为字段), 如果单击“工具→选项”菜单命令, 在打开的对话框中单击“重新计算”选项卡, 选中“工作簿选项”选项组中的“接受公式标志”选项, 公式就可以直接引用“列标志”了。例如“B2:B46”区域存放着学生的物理成绩, 而 B1 单元格已经输入了“物理”字样, 则求物理平均分的公式可以写成“=AVERAGE(物理)”。

需要特别说明的是, 创建好的名称可以被所有工作表引用, 而且引用时不需要在名称前面添加工作表名(这就是使用名称的主要优点), 因此名称引用实际上是一种绝对引用。但是公式引用“列标志”时的限制较多, 它只能在当前数据列的下方引用, 不能跨越工作表引用, 但是引用“列标志”的公式在一定条件下可以复制。从本质上讲, 名称和标志都是单元格引用的一种方式。因为它们不是文本, 使用时名称和标志都不能添加引号。

3. 函数输入方法

对 Excel 公式而言, 函数是其中的主要组成部分, 因此公式输入可以归结为函数输入的问题。

[1]. “插入函数”对话框

“插入函数”对话框是 Excel 输入公式的重要工具, 以公式“=SUM(Sheet2!A1:A6, Sheet3!B2:B9)”为例, Excel 输入该公式的具体过程是: 首先选中存放计算结果(即需要应用公式)的单元格, 单击编辑栏(或工具栏)中的“fx”按钮, 则表示公式开始的“=”出现在单元格和编辑栏, 然后在打开的“插入函数”对话框中的“选

择函数”列表找到“SUM”函数。如果你需要的函数不在里面，可以打开“或选择类别”下拉列表进行选择。最后单击“确定”按钮，打开“函数参数”对话框。对 SUM 函数而言，它可以使用从 number1 开始直到 number30 共 30 个参数。对上面的公式来说，首先应当把光标放在对话框的“number1”框中，单击工作簿中的“Sheet2!”工作表标签，“Sheet2!”即可自动进入其中，接着鼠标拖动选中你要引用的区域即可。接着用鼠标单击对话框的“number2”框，单击工作簿中的“Sheet3!”工作表标签，其 2 名称“Sheet3!”即可自动进入其中，再按相同方法选择要引用的单元格区域即可。

上述方法的最大优点就是引用的区域很准确，特别是三维引用时不容易发生工作表或工作簿名称输入错误的问题。

[2]. 编辑栏输入

如果你要套用某个现成公式，或者输入一些嵌套关系复杂的公式，利用编辑栏输入更加快捷。

首先选中存放计算结果的单元格；鼠标单击 Excel 编辑栏，按照公式的组成顺序依次输入各个部分，公式输入完毕后，单击编辑栏中的“输入”（即“√”）按钮（或回车）即可。手工输入时同样可以采取上面介绍的方法引用区域，以公式“=SUM(Sheet2!A1:A6,Sheet3!B2:B9)”为例，你可以先在编辑栏中输入“=SUM()”，然后将光标插入括号中间，再按上面介绍的方法操作就可以引用输入公式了。但是分隔引用之间的逗号必须用手工输入，而不能像“插入函数”对话框那样自动添加。在《电脑报 2003 年合订本附赠光盘》相应文章中，我们列出了所有 Excel2003 函数，供读者速查。

二. 函数速查一览

1. 数据库函数

[1]. DAVERAGE

用途：返回数据库或数据清单中满足指定条件的列中数值的平均值

语法：DAVERAGE(database, field, criteria)

参数：Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。Criteria 为一组包含给定条件的单元格区域。

[2]. DCOUNT

用途：返回数据库或数据清单的指定字段中，满足给定条件并且包含数字的单元格数目。

语法：DCOUNT(database, field, criteria)

参数：Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。Criteria 为一组包含给定条件的单元格区域。

[3]. DCOUNTA

用途：返回数据库或数据清单指定字段中满足给定条件的非空单元格数目。

语法: DCOUNTA(database, field, criteria)

参数: Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。
Criteria 为一组包含给定条件的单元格区域。

[4]. DGET

用途: 从数据清单或数据库中提取符合指定条件的单个值。

语法: DGET(database, field, criteria)

参数: Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。
Criteria 为一组包含给定条件的单元格区域。

[5]. DMAX

用途: 返回数据清单或数据库的指定列中, 满足给定条件单元格中的最大数值。

语法: DMAX(database, field, criteria)

参数: Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。
Criteria 为一组包含给定条件的单元格区域。

[6]. DMIN

用途: 返回数据清单或数据库的指定列中满足给定条件的单元格中的最小数字。

语法: DMIN(database, field, criteria)

参数: Database 构成列表或数据库的单元格区域。Field 指定函数所使用的数据列。
Criteria 为一组包含给定条件的单元格区域。

[7]. DPRODUCT

用途: 返回数据清单或数据库的指定列中, 满足给定条件单元格中数值乘积。

语法: DPRODUCT(database, field, criteria)

参数: 同上

[8]. DSTDEV

用途: 将列表或数据库的列中满足指定条件的数字作为一个样本, 估算样本总体的标准偏差。

语法: DSTDEV(database, field, criteria)

参数: 同上

[9]. DSTDEVP

参数: 将数据清单或数据库的指定列中, 满足给定条件单元格中的数字作为样本总体, 计算总体的标准偏差。

语法: DSTDEVP(database, field, criteria)

参数: 同上

[10]. DSUM

用途：返回数据清单或数据库的指定列中，满足给定条件单元格中的数字之和

语法：DSUM(database, field, criteria)

参数：同上

[11]. DVAR

用途：将数据清单或数据库的指定列中满足给定条件单元格中的数字作为一个样本，估算样本总体的方差。

语法：DVAR(database, field, criteria)

参数：同上

[12]. DVARP

用途：将数据清单或数据库的指定列中满足给定条件单元格中的数字作为样本总体，计算总体的方差。

语法：DVARP(database, field, criteria)

参数：同上

[13]. GETPIVOTDATA

用途：返回存储在数据透视表报表中的数据。如果报表中的汇总数据可见，则可以使用函数 GETPIVOTDATA 从数据透视表报表中检索汇总数据。

语法：GETPIVOTDATA(pivot_table, name)

参数：Data_field 为包含要检索的数据的数据字段的名称(放在引号中)。Pivot_table 在数据透视表中对任何单元格、单元格区域或定义的单元格区域的引用，该信息用于决定哪个数据数据透视表包含要检索的数据。Field1, Item1, Field2, Item2 为 1 到 14 对用于描述检索数据的字段名和项名称，可以任意次序排列。

2. 日期与时间函数

[1]. DATE

用途：返回代表特定日期的序列号。

语法：DATE(year, month, day)

参数：year 为一到四位，根据使用的日期系统解释该参数。默认情况下，Excel for Windows 使用 1900 日期系统，而 Excel for Macintosh 使用 1904 日期系统。Month 代表每年中月份的数字。如果所输入的月份大于 12，将从指定年份的一月份执行加法运算。Day 代表在该月份中第几天的数字。如果 day 大于该月份的最大天数时，将从指定月份的第一天开始往上累加。注意：Excel 按顺序的序列号保存日期，这样就可以对其进行计算。如果工作簿使用的是 1900 日期系统，则 Excel 会将 1900 年 1 月 1 日保存为序列号 1。同理，会将 1998 年 1 月 1 日保存为序列号 35796，因为该日期距离 1900 年 1 月 1 日为 35795 天。

实例：如果采用 1900 日期系统(Excel 默认)，则公式 “=DATE(2001, 1, 1)” 返回

36892。

[2]. DATEVALUE

用途：返回 date_text 所表示的日期的序列号。该函数的主要用途是将文字表示的日期转换成一个序列号。

语法：DATEVALUE(date_text)

参数：Date_text 是用 Excel 日期格式表示日期的文本。在使用 1900 日期系统中，date_text 必须是 1900 年 1 月 1 日到 9999 年 12 月 31 日之间的一个日期；而在 1904 日期系统中，date_text 必须是 1904 年 1 月 1 日到 9999 年 12 月 31 日之间的一个日期。如果 date_text 超出上述范围，则函数 DATEVALUE 返回错误值 #VALUE!。如果省略参数 date_text 中的年代，则函数 DATEVALUE 使用电脑系统内部时钟的当前年代，且 date_text 中的时间信息将被忽略。

实例：公式 “=DATEVALUE("2001/3/5")” 返回 36955，DATEVALUE("2-26") 返回 36948。

[3]. DAY

用途：返回用序列号(整数 1 到 31)表示的某日期的天数，用整数 1 到 31 表示。

语法：DAY(serial_number)

参数：Serial_number 是要查找的天数日期，它有多种输入方式：带引号的文本串(如 "1998/01/30")、序列号(如 1900 日期系统的 35825 表示的 1998 年 1 月 30 日)，以及其他公式或函数的结果(如 DATEVALUE("1998/1/30"))。

实例：公式 “=DAY("2001/1/27")” 返回 27，=DAY(35825) 返回 30，=DAY(DATEVALUE("2001/1/25")) 返回 25。

[4]. DAYS360

用途：按照一年 360 天的算法(每个月 30 天，一年共计 12 个月)，返回两日期间相差的天数。

语法：DAYS360(start_date, end_date, method)

参数：Start_date 和 end_date 是用于计算期间天数的起止日期。如果 start_date 在 end_date 之后，则 DAYS360 将返回一个负数。日期可以有多种输入方式：带引号的文本串(例如："1998/01/30")、序列号(例如：如果使用 1900 日期系统，则 35825 表示 1998 年 1 月 30 日)或其他公式或函数的结果(例如，DATEVALUE("1998/1/30"))。Method 是一个逻辑值，它指定了在计算中是采用欧洲方法还是美国方法。若为 FALSE 或忽略，则采用美国方法(如果起始日期是一个月的 31 日，则等于同月的 30 日。如果终止日期是一个月的 31 日，并且起始日期早于 30 日，则终止日期等于下一个月的 1 日，否则，终止日期等于本月的 30 日)。若为 TRUE 则采用欧洲方法(无论是起始日期还是终止日期为一个月的 31 号，都将等于本月的 30 号)。

实例：公式 “=DAYS360("1998/2/1", "2001/2-1")” 返回 1080。

[5]. EDATE

用途：返回指定日期(start_date)之前或之后指定月份的日期序列号。

语法：EDATE(start_date, months)

参数: Start_date 参数代表开始日期, 它有多种输入方式: 带引号的文本串(例如: "1998/01/30")、序列号(如 35825 表示 1998 年 1 月 30 日)或其他公式或函数的结果(例如: DATEVALUE("1998/1/30"))。Months 为在 start_date 之前或之后的月份数, 未来日期用正数表示, 过去日期用负数表示。

实例: 公式 “=EDATE("2001/3/5", 2)” 返回 37016 即 2001 年 5 月 5 日, =EDATE("2001/3/5", -6)返回 36774 即 2000 年 9 月 5 日。

[6]. EOMONTH

用途: 返回 start-date 之前或之后指定月份中最后一天的序列号。

语法: EOMONTH(start_date, months)

参数: Start_date 参数代表开始日期, 它有多种输入方式: 带引号的文本串(如 "1998/01/30")、序列号(如 1900 日期系统中的 35825)或其他公式或函数的结果(如 DATEVALUE("1998/1/30"))。Month 为 start_date 之前或之后的月份数, 正数表示未来日期, 负数表示过去日期。

实例: 公式 “=EOMONTH("2001/01/01", 2)” 返回 36981 即 2001 年 3 月 31 日, =EOMONTH("2001/01/01", -6)返回 36738 即 2000 年 7 月 31 日。

[7]. HOUR

用途: 返回时间值的小时数。即介于 0(12:00 A.M.)到 23(11:00 P.M.)之间的一个整数。

语法: HOUR(serial_number)

参数: Serial_number 表示一个时间值, 其中包含着要返回的小时数。它有多种输入方式: 带引号的文本串(如 "6:45 PM")、十进制数(如 0.78125 表示 6:45PM)或其他公式或函数的结果(如 TIMEVALUE("6:45PM"))。

实例: 公式 “=HOUR("3:30:30PM”)” 返回 15, =HOUR(0.5)返回 12 即 12:00:00AM, =HOUR(29747.7)返回 16。

[8]. MINUTE

用途: 返回时间值中的分钟, 即介于 0 到 59 之间的一个整数。

语法: MINUTE(serial_number)

参数: Serial_number 是一个时间值, 其中包含着要查找的分钟数。时间有多种输入方式: 带引号的文本串(如 "6:45 PM")、十进制数(如 0.78125 表示 6:45 PM)或其他公式或函数的结果(如 TIMEVALUE("6:45 PM"))。

实例: 公式 “=MINUTE("15:30:00”)” 返回 30, =MINUTE(0.06)返回 26, =MINUTE(TIMEVALUE("9:45 PM"))返回 45。

[9]. MONTH

用途: 返回以序列号表示的日期中的月份, 它是介于 1(一月)和 12(十二月)之间的整数。

语法: MONTH(serial_number)

参数: Serial_number 表示一个日期值, 其中包含着要查找的月份。日期有多种输入方式: 带引号的文本串(如 "1998/01/30")、序列号(如表示 1998 年 1 月 30 日的 35825)或其他公式或函数的结果(如 DATEVALUE("1998/1/30"))等。

实例：公式“=MONTH("2001/02/24")”返回 2，=MONTH(35825)返回 1，
=MONTH(DATEVALUE("2000/6/30"))返回 6。

[10]. NETWORKDAYS

用途：返回参数 start-data 和 end-data 之间完整的工作日(不包括周末和专门指定的假期)数值。

语法：NETWORKDAYS(start_date, end_date, holidays)

参数：Start_date 代表开始日期，End_date 代表终止日；Holidays 是表示不在工作日历中的一个或多个日期所构成的可选区域，法定假日以及其他非法定假日。此数据清单可以是包含日期的单元格区域，也可以是由代表日期的序列号所构成的数组常量。函数中的日期有多种输入方式：带引号的文本串(如"1998/01/30")、序列号(如使用 1900 日期系统的 35825)或其他公式或函数的结果(如 DATEVALUE("1998/1/30"))。注意：该函数只有加载“分析工具库”以后方能使用。

[11]. NOW

用途：返回当前日期和时间所对应的序列号。

语法：NOW()

参数：

实例：如果正在使用的是 1900 日期系统，而且计算机的内部时钟为 2001-1-28 12:53，
则公式“=NOW()”返回 36919.54。

[12]. SECOND

用途：返回时间值的秒数(为 0 至 59 之间的一个整数)。

语法：SECOND(serial_number)

参数：Serial_number 表示一个时间值，其中包含要查找的秒数。关于时间的输入方式见上文的有关内容。

实例：公式“=SECOND("3:30:26PM")”返回 26，=SECOND(0.016)返回 2。

[13]. TIME

用途：返回某一特定时间的小数值，它返回的小数值从 0 到 0.99999999 之间，代表 0:00:00(12:00:00A.M)到 23:59:59(11:59:59P.M)之间的时间。

语法：TIME(hour, minute, second)

参数：Hour 是 0 到 23 之间的数，代表小时；Minute 是 0 到 59 之间的数，代表分；Second 是 0 到 59 之间的数，代表秒。

实例：公式“=TIME(12, 10, 30)”返回序列号 0.51，等价于 12:10:30PM。=TIME(9, 30, 10)返回序列号 0.40，等价于 9:30:10AM。=TEXT(TIME(23, 18, 14), "h:mm:ssAM/PM")返回“11:18:14PM”。

[14]. TIMEVALUE

用途：返回用文本串表示的时间小数值。该小数值为从 0 到 0.99999999 的数值，代表从 0:00:00(12:00:00AM)到 23:59:59(11:59:59PM)之间的时间。

语法: TIMEVALUE(time_text)

参数: Time_text 是一个用 Excel 时间格式表示时间的文本串(如"6:45PM"和"18:45"等)。

实例: 公式 " =TIMEVALUE("3:30AM") " 返回 0.145833333 , 公式 " =TIMEVALUE("2001/1/26 6:35AM") " 返回 0.274305556。

[15]. TODAY

用途: 返回系统当前日期的序列号。

语法: TODAY()

参数:

实例: 公式 " =TODAY() " 返回 2001-8-28(执行公式时的系统时间)。

[16]. WEEKDAY

用途: 返回某日期的星期数。在默认情况下, 它的值为 1(星期天)到 7(星期六)之间的一个整数。

语法: WEEKDAY(serial_number, return_type)

参数: Serial_number 是要返回日期数的日期, 它有多种输入方式: 带引号的文本串(如"2001/02/26")、序列号(如 35825 表示 1998 年 1 月 30 日)或其他公式或函数的结果(如 DATEVALUE("2000/1/30"))。Return_type 为确定返回值类型的数字, 数字 1 或省略则 1 至 7 代表星期天到数星期六, 数字 2 则 1 至 7 代表星期一到星期天, 数字 3 则 0 至 6 代表星期一到星期天。

实例: 公式 " =WEEKDAY("2001/8/28" , 2) " 返回 2(星期二) , =WEEKDAY("2003/02/23", 3)返回 6(星期日)。

[17]. WEEKNUM

用途: 返回一个数字, 该数字代表一年中的第几周。

语法: WEEKNUM(serial_num, return_type)

参数: Serial_num 代表一周中的日期。应使用 DATE 函数输入日期, 或者将日期作为其他公式或函数的结果输入。Return_type 为一数字, 确定星期计算从哪一天开始。默认值为 1。

[18]. WORKDAY

用途: 返回某日期(起始日期)之前或之后相隔指定工作日(不包括周末和专门指定的假日)的某一日期, 并扣除周末或假日。

语法: WORKDAY(start_date, days, holidays)

参数: Start_date 为开始日期; Days 为 Start_date 之前

或之后不含周末及节假日的天数; Days 是正值将产生未来日 6 期、负值产生过去日期; Holidays 为可选的数据清单, 表示需要从工作日历中排除的日期值(如法定假日或非法定假日)。此清单可以是包含日期的单元格区域, 也可以是由代表日期的序列号所构成的数组常量。日期有多种输入方式: 带引号的文本串(如"1998/01/30")、序列号(如 1900 日期系统时的 35825 表示 1998 年 1 月 30 日)或其他公式或函数的结果(例如 DATEVALUE("1998/1/30"))。

[19]. YEAR

用途：返回某日期的年份。其结果为 1900 到 9999 之间的一个整数。

语法：YEAR(serial_number)

参数：Serial_number 是一个日期值，其中包含要查找的年份。日期有多种输入方式：带引号的文本串(例如"1998/01/30")、序列号(例如，如果使用 1900 日期系统则 35825 表示 1998 年 1 月 30 日)或其他公式或函数的结果(例如 DATEVALUE("1998/1/30"))

实例：公式 “=YEAR("2000/8/6")”返回 2000”，=YEAR("2003/05/01")返回 2003，=YEAR(35825)返回 1998。

[20]. YEARFRAC

用途：返回 start_date 和 end_date 之间的天数占全年天数的百分比。

语法：YEARFRAC(start_date, end_date, basis)

参数：Start_date 表示开始日期，End_date 代表结束日期。函数中的日期有多种输入方式：带引号的文本串(如"1998/01/30")、序列号(如 35829 表示 1900 日期系统中的 1998 年 1 月 30 日)，或其他公式或函数的结果(例如:DATEVALUE("1998/1/30"))。Basis 表示日计数基准类型，其中 0 或省略为 US(NASD)30/360, 1 实际天数/实际天数, 2 实际天数/360, 3 实际天数/365, 4 欧洲 30/360。

实例：公式 “=YEARFRAC("2001/01/31", "2001/06/30", 0)” 返回 0.416666667，YEARFRAC("2001/01/25", "2001/09/27")返回 0.67222。

3. 外部函数

[1]. EUROCONVERT

用途：将数字转换为欧元形式，将数字由欧元形式转换为欧盟成员国货币形式，或利用欧元作为中间货币将数字由某一欧盟成员国货币转化为另一欧盟成员国货币的形式(三角转换关系)。

语 法：EUROCONVERT(number, source, target, full_precision, triangulation_precision)

参数：Number 为要转换的货币值，或对包含该值的单元格的引用。Source 是由三个字母组成的字符串，或对包含字符串的单元格的引用，该字符串对应于源货币的 ISO 代码。EUROCONVERT 函数中可以使用下列货币代码：国家/地区基本货币单位 ISO 代码比利时法郎 BEF 卢森堡法郎 LUF 德国德国马克 DEM 西班牙西班牙比塞塔 ESP 法国法郎 FRF 爱尔兰爱尔兰磅 IEP 意大利里拉 ITL 荷兰荷兰盾 NLG 奥地利奥地利先令 ATS 葡萄牙埃斯库多 PTE 芬兰芬兰马克 FIM 希腊德拉克马 GRD 欧盟成员国欧元 EUR。

[2]. SQL.REQUEST

用途：与外部数据源连接，从工作表运行查询，然后 SQL.REQUEST 将查询结果以数组的形式返回，而无需进行宏编程。

语法：SQL.REQUEST(connection_string, output_ref, driver_prompt, query_text, col_names_logical)

参数: Connection_string 提供信息, 如数据源名称、用户 ID 和密码等。Output_ref 对用于存放完整的连接字符串的单元格的引用。Driver_prompt 指定驱动程序对话框何时显示以及何种选项可用。Column_names_logical 指示是否将列名作为结果的第一行返回。如果要将列名作为结果的第一行返回, 请将该参数设置为 TRUE。如果不需要将列名返回, 则设置为 FALSE。如果省略 column_names_logical, 则 SQL.REQUEST 函数不返回列名。

4. 工程函数

[1]. BESSELI

用途: 返回修正 Bessel 函数值, 它与用纯虚数参数运算时的 Bessel 函数值相等。

语法: BESSELI(x, n)

参数: X 为参数值。N 为函数的阶数。如果 n 非整数, 则截尾取整。

[2]. BESSELJ

用途: 返回 Bessel 函数值。

语法: BESSELJ(x, n)

参数: 同上

[3]. BESSELK

用途: 返回修正 Bessel 函数值, 它与用纯虚数参数运算时的 Bessel 函数值相等。

语法: BESSELK(x, n)

参数: 同上

[4]. BESSELY

用途: 返回 Bessel 函数值, 也称为 Weber 函数或 Neumann 函数。

语法: BESSELY(x, n)

参数: 同上

[5]. BIN2DEC

用途: 将二进制数转换为十进制数。

语法: BIN2DEC(number)

参数: Number 待转换的二进制数。Number 的位数不能多于 10 位(二进制位), 最高位为符号位, 后 9 位为数字位。负数用二进制数补码表示。

[6]. BIN2HEX

用途: 将二进制数转换为十六进制数。

语法: BIN2HEX(number, places)

参数: Number 为待转换的二进制数。Number 的位数不能多于 10 位(二进制位), 最高位为符号位, 后 9 位为数字位。负数用二进制数补码表示; Places 为所要使用的字符

数。如果省略 places，函数 DEC2BIN 用能表示此数的最少字符来表示。

[7]. BIN2OCT

用途：将二进制数转换为八进制数。

语法：BIN2OCT(number, places)

参数：Number 为待转换的二进制数；Places 为所要使用的字符数。

[8]. COMPLEX

用途：将实系数及虚系数转换为 $x+yi$ 或 $x+yj$ 形式的复数。

语法：COMPLEX(real_num, i_num, suffix)

参数：Real_num 为复数的实部，I_num 为复数的虚部，Suffix 为复数中虚部的后缀，省略时则认为它为 i。

[9]. CONVERT

用途：将数字从一个度量系统转换到另一个度量系统中。

语法：CONVERT(number, from_unit, to_unit)

参数：Number 是以 from_units 为单位的需要进行转换的数值。From_unit 是数值 number 的单位。To_unit 是结果的单位。

[10]. DEC2BIN

用途：将十进制数转换为二进制数。

语法：DEC2BIN(number, places)

参数：Number 是待转换的十进制数。Places 是所要使用的字符数，如果省略 places，函数 DEC2OCT 用能表示此数的最少字符来表示。

[11]. DEC2HEX

用途：将十进制数转换为十六进制数。

语法：DEC2HEX(number, places)

参数：Number 为待转换的十进制数。如

参数 number 是负数，则省略 places。Places 是所要使用的字符数。

[12]. DEC2OCT

用途：将十进制数转换为八进制数。

语法：DEC2OCT(number, places)

参数：Number 为待转换的十进制数。如果参数 number 是负数，则省略 places。Places 是所要使用的字符数。

[13]. DELTA

用途：测试两个数值是否相等。如果 number1=number2，则返回 1，否则返回 0

语法：DELTA(number1, number2)

参数: Number1 为第一个参数。Number2 为第二个参数。如果省略, 假设 Number2 的值为零。

[14]. ERF

用途: 返回误差函数在上下限之间的积分。

语法: ERF(lower_limit, upper_limit)

参数: Lower_limit 是 ERF 函数的积分下限。Upper_limit 是 ERF 函数的积分上限。如果省略, ERF 将在零到下限之间进行积分。

[15]. ERFC

用途: 返回从 x 到 ∞ (无穷) 积分的 ERF 函数的余误差函。

语法: ERFC(x)

参数: x 是 ERF 函数积分的下限。

[16]. GESTEP

用途: 如果 Number 大于等于 step, 返回 1, 否则返回 0。使用该函数可筛选数据。

语法: GESTEP(number, step)

参数: Number 是待测试的数值。Step 是阈值。如果省略 step, 则函数 GESTEP 假设其为零。

[17]. HEX2BIN

用途: 将十六进制数转换为二进制数。

语法: HEX2BIN(number, places)

参数: Number 是待转换的十六进制数, Places 是所要使用的字符数。

[18]. HEX2DEC

用途: 将十六进制数转换为十进制数。

语法: HEX2DEC(number)

参数: Number 是待转换的十六进制数

参数 number 的位数不能多于 10 位(40 位二进制), 最高位为符号位, 其余 39 位是数字位。负数用二进制的补码表示。

[19]. HEX2OCT

用途: 将十六进制数转换为八进制数。

语法: HEX2OCT(number, places)

参数: Number 是待转换的十六进制数, Places 是所要使用的字符数。

[20]. IMABS

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的绝对值(模)。

语法: IMABS(inumber)

参数: Inumber?为需要计算其绝对值的复数。

[21]. IMAGINARY

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的虚系数。

语法: IMAGINARY(inumber)

参数: Inumber?为需要计算其虚系数的复数。

[22]. IMARGUMENT

用途: 返回以弧度表示的角。

语法: IMARGUMENT(inumber)

参数: Inumber 为用来计算角度值的复数。

[23]. MCONJUGATE

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的共轭复数。

语法: IMCONJUGATE(inumber)

参数: Inumber 为需要计算其共轭数的复数。

[24]. IMCOS

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的余弦。

语法: MCOS(inumber)

参数: Inumber 为需要计算其余弦值的复数。

[25]. IMDIV

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的两个复数的商。

语法: IMDIV(inumber1, inumber2)

参数: Inumber1 为复数分子(被除数), Inumber2 为复数分母(除数)。

[26]. IMEXP

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的指数。

语法: IMEXP(inumber)

参数: Inumber?为需要计算其指数的复数。

[27]. IMLN

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的自然对数。

语法: IMLN(inumber)

参数: Inumber 为需要计算其自然对数的复数。9

[28]. IMLOG10

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的常用对数(以 10 为底数)。

语法: IMLOG10(inumber)

参数: Inumber?为需要计算其常用对数的复数。

[29]. IMLOG2

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的以 2 为底数的对数。

语法: IMLOG2(inumber)

参数: Inumber 为需要计算以 2 为底数的对数值的复数。

[30]. IMPOWER

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的 n 次幂。

语法: IMPOWER(inumber, number)

参数: Inumber 为需要计算其幂值的复数, Number 为需要计算的幂次。

[31]. IMPRODUCT

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的 2 至 29 个复数的乘积。

语法: IMPRODUCT(inumber1, inumber2, ...)

参数: Inumber1, inumber2, .为 1 到 29 个用来相乘的复数。

[32]. IMREAL

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的实系数。

语法: IMREAL(inumber)

参数: Inumber?为需要计算其实系数的复数。

[33]. IMSIN

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的正弦值。

语法: IMSIN(inumber)

参数: Inumber?为需要计算其正弦的复数。

[34]. IMSQRT

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的复数的平方根。

语法: IMSQRT(inumber)

参数: Inumber 为需要计算其平方根的复数。

[35]. IMSUB

用途: 返回以 $x+yi$ 或 $x+yj$ 文本格式表示的两个复数的差。

语法: IMSUB(inumber1, inumber2)

参数: Inumber1 是被减(复)数, Inumber2 是为减(复)数。

[36]. IMSUM

用途：返回以 $x+yi$ 或 $x+yj$ 文本格式表示的两个或多个复数的和。

语法：IMSUM(inumber1, inumber2, ...)

参数：Inumber1, inumber2, ...为 1 到 29 个需要相加的复数。

[37]. OCT2BIN

用途：将八进制数转换为二进制数。

语法：OCT2BIN(number, places)

参数：Number 是待转换的八进制数。Places 是所要使用的字符数。

[38]. OCT2DEC

用途：将八进制数转换为十进制数。

语法：OCT2DEC(number)

参数：Number 是待转换的八进制数。

[39]. OCT2HEX

用途：将八进制数转换为十六进制数。

语法：OCT2HEX(number, places)

参数：Number 是待转换的八进制数。Places 是所要使用的字符数。

5. 财务函数

[1]. ACCRINT

用途：返回定期付息有价证券的应计利息。

语法：ACCRINT(issue, first_interest, settlement, rate, par, frequency, basis)

参数：Issue 为有价证券的发行日，First_interest 是证券的起息日，Settlement 是证券的成交日(即发行日之后证券卖给购买者的日期)，Rate 为有价证券的年息票利率，Par 为有价证券的票面价值(如果省略 par，函数 ACCRINT 将 par 看作\$1000)，Frequency 为年付息次数(如果按年支付，10frequency=1；按半年期支付，frequency=2；按季支付，frequency=4)。

[2]. ACCRINTM

用途：返回到期一次性付息有价证券的应计利息。

语法：ACCRINTM(issue, maturity, rate, par, basis)

参数：Issue 为有价证券的发行日，Maturity 为有价证券的到期日，Rate 为有价证券的年息票利率，Par 为有价证券的票面价值，Basis 为日计数基准类型(0 或省略时为 30/360，1 为实际天数/实际天数，2 为实际天数/360，3 为实际天数/365，4 为欧洲 30/360)。

[3]. AMORDEGRC

用途：返回每个会计期间的折旧值。

语法：AMORDEGRC(cost, date_purchased, first_period, salvage, period, rate, basis)

参数：Cost 为资产原值，Date_purchased 为购入资产的日期，First_period 为第一个期间结束时的日期，Salvage 为资产在使用寿命结束时的残值，Period 是期间，Rate 为折旧率，Basis 是所使用的年基准(0 或省略时为 360 天，1 为实际天数，3 为一年 365 天，4 为一年 360 天)。

[4]. AMORLINC

用途：返回每个会计期间的折旧值，该函数为法国会计系统提供。如果某项资产是在会计期间内购入的，则按线性折旧法计算。

语法：AMORLINC(cost, date_purchased, first_period, salvage, period, rate, basis)

参数：Date_purchased 为购入资产的日期，First_period 为第一个期间结束时的日期，Salvage 为资产在使用寿命结束时的残值，Period 为期间，Rate 为折旧率，Basis 为所使用的年基准(0 或省略时为 360 天，1 为实际天数，3 为一年 365 天，4 为一年 360 天)。

[5]. COUPDAYBS

用途：返回当前付息期内截止到成交日的天数。

语法：COUPDAYBS(settlement, maturity, frequency, basis)

参数：Settlement 是证券的成交日(即发行日之后证券卖给购买者的日期)，Maturity 为有价证券的到期日，Frequency 为年付息次数(如果按年支付，frequency=1；按半年期支付，frequency=2；按季支付，frequency=4)，Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[6]. COUPDAYS

用途：返回成交日所在的付息期的天数。

语法：COUPDAYS(settlement, maturity, frequency, basis)

参数：Settlement 是证券的成交日(即发行日之后证券卖给购买者的日期)，Maturity 为有价证券的到期日(即有价证券有效期截止时的日期)，Frequency 为年付息次数(如果按年支付，frequency=1；按半年期支付，frequency=2；按季支付，frequency=4)，Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[7]. COUPDAYSNCR

用途：返回从成交日到下一付息日之间的天数。

语法：COUPDAYSNCR(settlement, maturity, frequency, basis)

参数：Settlement 是证券的成交日，Maturity 为有价证券的到期日，Frequency 为

年付息次数(如果按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[8]. COUPNUM

用途: 返回成交日和到期日之间的利息应付次数, 向上取整到最近的整数。

语法: COUPNUM(settlement, maturity, frequency, basis)

参数: 同上

[9]. COUPPCD

用途: 用途: 返回成交日之前的上一付息日的日期。

语法: COUPPCD(settlement, maturity, frequency, basis)

参数: 同上

[10]. CUMIPMT

用途: 返回一笔贷款在给定的 start-period 到 end-period 期间累计偿还的利息数额。

语法: CUMIPMT(rate, nper, pv, start_period, end_period, type)

参数: Rate 为利率, Nper 为总付款期数, Pv 为现值, Start_period 为计算中的首期(付款期数从 1 开始计数), End_period 为计算中的末期, Type 为付款时间类型(0(零)为期末付款, 1 为期初付款)。

[11]. CUMPRINC

用途: 返回一笔贷款在给定的 start-period 到 end-period 期间累计偿还的本金数额。

语法: CUMPRINC(rate, nper, pv, start_period, end_period, type)

参数: Rate 为利率, Nper 为总付款期数, Pv 为现值 Start_period 为计算中的首期(付款期数从 1 开始计数), End_period 为计算中的末期, Type 为付款时间类型(0(零)为期末付款, 1 为期初付款)。

[12]. DB

用途: 使用固定余额递减法, 计算一笔资产在给定期间的折旧值。

语法: DB(cost, salvage, life, period, month)

参数: Cost 为资产原值, Salvage 为资产在折旧期末的价值(也称为资产残值), Life 为折旧期限(有时也称作资产的使用寿命), Period 为需要计算折旧值的期间。Period 必须使用与 life 相同的单位, Month 为第一年的月份数(省略时假设为 12)。

[13]. DDB

用途: 使用双倍余额递减法或其他指定方法, 计算一笔资产在给定期间的折旧值。

语法: DDB(cost, salvage, life, period, factor)

参数: Cost 为资产原值, Salvage 为资产在折旧期末的价值(也称为资产残值), Life 为折旧期限(有时也称作资产的使用寿命), Period 为需要计算折旧值的期间。Period 必须

使用与 life 相同的单位，Factor 为余额递

[14]. DISC

用途：返回有价证券的贴现率。

语法：DISC(settlement, maturity, pr, redemption, basis)

参数：Settlement 是证券的成交日(即在发行日之后，证券卖给购买者的日期)，Maturity 为有价证券的到期日，Pr 为面值\$100 的有价证券的价格，Redemption 为面值\$100 的有价证券的清偿价值，Basis 为日计数基准类型(0 或省略为 30/360，1 为实际天数/实际天数，2 为实际天数/360，3 为实际天数/365，4 为欧洲 30/360)。

[15]. DOLLARDE

用途：将按分数表示的价格转换为按小数表示的价格，如证券价格，转换为小数表示的数字。

语法：DOLLARDE(fractional_dollar, fraction)

参数：Fractional_dollar 以分数表示的数字，Fraction 分数中的分母(整数)。

[16]. DOLLARFR

用途：将按小数表示的价格转换为按分数表示的价格。

语法：DOLLARFR(decimal_dollar, fraction)

参数：Decimal_dollar 为小数，Fraction 分数中的分母(整数)。

[17]. DURATION

用途：返回假设面值\$100 的定期付息有价证券的修正期限。期限定义为一系列现金流现值的加权平均值，用于计量债券价格对于收益率变化的敏感程度。

语法：DURATION(settlement, maturity, couponyld, frequency, basis)

参数：Settlement 是证券的成交日，Maturity 为有价证券的到期日，Coupon 为有价证券的年息票利率，Yld 为有价证券的年收益率，Frequency 为年付息次数(如果按年支付，frequency=1；按半年期支付，frequency=2；按季支付，frequency=4)，Basis 日计数基准类型(0 或省略为 30/360，1 为实际天数/实际天数，2 为实际天数/360，3 为实际天数/365，4 为欧洲 30/360)。

[18]. EFFECT

用途：利用给定的名义年利率和一年中的复利期次，计算实际年利率。

语法：EFFECT(nominal_rate, npery)

参数：Nominal_rate 为名义利率，Npery 为每年的复利期数。

[19]. FV

用途：基于固定利率及等额分期付款方式，返回某项投资的未来值。

语法：FV(rate, nper, pmt, pv, type)

参数：Rate 为各期利率，Nper 为总投资期(即该项投资的付款期总数)，Pmt 为各期

所应支付的金额, Pv 为现值(即从该项投资开始计算时已经入帐的款项, 或一系列未来付款的当前值的累积和, 也称为本金), Type 为数字 0 或 1(0 为期末, 1 为期初)。

[20]. FVSCCHEDULE

用途: 基于一系列复利返回本金的未来值, 用于计算某项投资在变动或可调利率下的未来值。

语法: FVSCCHEDULE(principal, schedule)

参数: Principal 为现值, Schedule 为利率数组。

[21]. INTRATE

用途: 返回一次性付息证券的利率。

语法: INTRATE(settlement, maturity, investment, redemption, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Investment 为有价证券的投资额, Redemption 为有价证券到期时的清偿价值, Basis 日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[22]. IPMT

用途: 基于固定利率及等额分期付款方式, 返回投资或贷款在某一给定期限内的利息偿还额。

语法: IPMT(rate, per, nper, pv, fv, type)

参数: Rate 为各期利率, Per 用于计算其利息数额的期数(1 到 nper 之间), Nper 为总投资期, Pv 为现值(本金), Fv 为未来值(最后一次付款后的现金余额。如果省略 fv, 则假设其值为零), Type 指定各期的付款时间是在期初还是期末(0 为期末, 1 为期初)。

[23]. IRR

用途: 返回由数值代表的一组现金流的内部收益率。

语法: IRR(values, guess)

参数: Values 为数组或单元格的引用, 包含用来计算返回的内部收益率的数字。Guess 为对函数 IRR 计算结果的估计值。

[24]. ISPMT

用途: 计算特定投资期内要支付的利息。

语法: ISPMT(rate, per, nper, pv)

参数: Rate 为投资的利率, Per 为要计算利息的期数(在 1 到 nper 之间), Nper 为投资的总支付期数, Pv 为投资的当前值(对于贷款来说 pv 为贷款数额)。

[25]. MDURATION

用途: 返回假设面值\$100 的有价证券的 Macauley 修正期限。

语法: MDURATION(settlement, maturity, coupon, yld, frequency, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Coupon 为有价证券的年息票利率, Yld 为有价证券的年收益率, Frequency 为年付息次数(如果按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[26]. MIRR

用途: 返回某一期限内现金流的修正内部收益率。

语法: MIRR(values, finance_rate, reinvest_rate)

参数: Values 为一个数组或对包含数字的单元格的引用(代表着各期的一系列支出及收入, 其中必须至少包含一个正值和一个负值, 才能计算修正后的内部收益率), Finance_rate 为现金流中使用的资金支付的利率, Reinvest_rate 为将现金流再投资的收益率。

[27]. NOMINAL

用途: 基于给定的实际利率和年复利期数, 返回名义年利率。

语法: NOMINAL(effect_rate, npery)

参数: Effect_rate 为实际利率, Npery 为每年的复利期数。

[28]. NPER

用途: 基于固定利率及等额分期付款方式, 返回某项投资(或贷款)的总期数。

语法: NPER(rate, pmt, pv, fv, type)

参数: Rate 为各期利率, Pmt 为各期所应支付的金额, Pv 为现值(本金), Fv 为未来值(即最后一次付款后希望得到的现金余额), Type 可以指定各期的付款时间是在期初还是期末(0 为期末, 1 为期初)。

[29]. NPV

用途: 通过使用贴现率以及一系列未来支出(负值)和收入(正值), 返回一项投资的净现值。

语法: NPV(rate, value1, value2, ...)

参数: Rate 为某一期间的贴现率, Value1, value2, ...为 1 到 29 个参数, 代表支出及收入。

[30]. ODDFPRICE

用途: 返回首期付息日不固定的面值\$100 的有价证券的价格。

语法: ODDFPRICE(settlement, maturity, issue, first_coupon, rate, yld, redemption, frequency, basis)

参数: Settlement 为证券的成交日, Maturity 为有价证

券的到期日, Issue 为有价证券的发行日, First_coupon 为有价证券的首期付息日, Rate 为有价证券的利率, Yld 为有价证券的年收益率, Redemption 为面值\$100 的有价证券的清偿价值, Frequency 为年付息次数(如果按年支付, frequency=1; 按半年期支

付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[31]. ODDFYIELD

用途: 返回首期付息日不固定的有价证券(长期或短期)的收益率。

语法: ODDFYIELD(settlement, maturity, issue, first_coupon, rate, pr, redemption, frequency, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Issue 为有价证券的发行日, First_coupon 为有价证券的首期付息日, Rate 为有价证券的利率, Pr 为有价证券的价格, Redemption 为面值\$100 的有价证券的清偿价值, Frequency 为年付息次数(按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[32]. ODDLPRICE

用途: 返回末期付息日不固定的面值\$100 的有价证券(长期或短期)的价格。

语法: ODDLPRICE(settlement, maturity, last_interest, rate, yld, redemption, frequency, basis)

参数: Settlement 为有价证券的成交日, Maturity 为有价证券的到期日, Last_interest 为有价证券的末期付息日, Rate 为有价证券的利率, Yld 为有价证券的年收益率, Redemption 为面值\$100 的有价证券的清偿价值, Frequency 为年付息次数(如果按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[33]. ODDLYIELD

用途: 返回末期付息日不固定的有价证券(长期或短期)的收益率。

语法: ODDLYIELD(settlement, maturity, last_interest, rate, pr, redemption, frequency, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Last_interest 为有价证券的末期付息日, Rate 为有价证券的利率, Pr 为有价证券的价格, Redemption 为面值\$100 的有价证券的清偿价值, Frequency 为年付息次数(如果按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[34]. PMT

用途: 基于固定利率及等额分期付款方式, 返回贷款的每期付款额。

语法: PMT(rate, nper, pv, fv, type)

参数: Rate 贷款利率, Nper 该项贷款的付款总数, Pv 为现值(也称为本金), Fv 为未来值(或最后一次付款后希望得到的现金余额), Type 指定各期的付款时间是在期初还是期末(1 为期初, 0 为期末)。

[35]. PPMT

用途: 基于固定利率及等额分期付款方式, 返回投资在某一给定期间的本金偿还额。

语法: PPMT(rate, per, nper, pv, fv, type)

参数: Rate 为各期利率, Per 用于计算其本金数额的期数(介于 1 到 nper 之间), Nper 为总投资期(该项投资的付款期总数), Pv 为现值(也称为本金), Fv 为未来值, Type 指定各期的付款时间是在期初还是期末(1 为期初。0 为期末)。

[36]. PRICE

用途: 返回定期付息的面值\$100 的有价证券的价格。

语法: PRICE(settlement, maturity, rate, yld, redemption, frequency, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Rate 为有价证券的年息票利率, Yld 为有价证券的年收益率, Redemption 为面值\$100 的有价证券的清偿价值, Frequency 为年付息次数(如果按年支付, frequency=1; 按半年期支付, frequency=2; 按季支付, frequency=4), Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[37]. PRICEDISC

用途: 返回折价发行的面值\$100 的有价证券的价格。

语法: PRICEDISC(settlement, maturity, discount, redemption, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Discount 为有价证券的贴现率, Redemption 为面值\$100 的有价证券的清偿价值, Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[38]. PRICEMAT

用途: 返回到期付息的面值\$100 的有价证券的价格。

语法: PRICEMAT(settlement, maturity, issue, rate, yld, basis)

参数: Settlement 为证券的成交日, Maturity 为有价证券的到期日, Issue 为有价证券的发行日(以时间序列号表示), Rate 为有价证券在发行日的利率, Yld 为有价证券的年收益率, Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[39]. PV

用途: 返回投资的现值(即一系列未来付款的当前值的累积和), 如借入方的借入款即为贷出方贷款的现值。

语法: PV(rate, nper, pmt, fv, type)

参数: Rate 为各期利率, Nper 为总投资(或贷款)期数, Pmt 为各期所应支付的金额, Fv 为未来值, Type 指定各期的付款时间是在期初还是期末(1 为期初。0 为期末)。

[40]. RATE

用途：返回年金的各期利率。函数 RATE 通过迭代法计算得出，并且可能无解或有多个解。

语法：RATE(nper, pmt, pv, fv, type, guess)

参数：Nper 为总投资期(即该项投资的付款期总数)，Pmt 为各期付款额，Pv 为现值(本金)，Fv 为未来值，Type 指定各期的付款时间是在期初还是期末(1 为期初。0 为期末)。

[41]. RECEIVED

用途：返回一次性付息的有价证券到期收回的金额。

语法：RECEIVED(settlement, maturity, investment, discount, basis)

参数：Settlement 为证券的成交日，Maturity 为有价证券的到期日，Investment 为有价证券的投资额，Discount 为有价证券的贴现率，Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[42]. SLN

用途：返回某项资产在一个期间中的线性折旧值。

语法：SLN(cost, salvage, life)

参数：Cost 为资产原值，Salvage 为资产在折旧期末的价值(也称为资产残值)，Life 为折旧期限(有时也称作资产的使用寿命)。

[43]. SYD

用途：返回某项资产按年限总和折旧法计算的指定期间的折旧值。

语法：SYD(cost, salvage, life, per)

参数：Cost 为资产原值，Salvage 为资产在折旧期末的价值(也称为资产残值)，Life 为折旧期限(有时也称作资产的使用寿命)，Per 为期间(单位与 life 相同)。

[44]. TBILLEQ

用途：返回国库券的等效收益率。

语法：TBILLEQ(settlement, maturity, discount)

参数：Settlement 为国库券的成交日(即在发行日之后，国库券卖给购买者的日期)，Maturity 为国库券的到期日，Discount 为国库券的贴现率。

[45]. TBILLPRICE

用途：返回面值\$100 的国库券的价格。

语法：TBILLPRICE(settlement, maturity, discount)

参数：Settlement 为国库券的成交日，Maturity 为国库券的到期日，Discount 为国库券的贴现率。

[46]. TBILLYIELD

用途：返回国库券的收益率。

语法：TBILLYIELD(settlement, maturity, pr)

参数：Settlement 为国库券的成交日，Maturity 为国库券的到期日，Pr 为面值\$100 的国库券的价格。

[47]. VDB

用途：使用双倍余额递减法或其他指定的方法，返回指定的任何期间内(包括部分期间)的资产折旧值。

语法：VDB(cost, salvage, life, start_period, end_period, factor, no_switch)

参数：Cost 为资产原值，Salvage 为资产在折旧期末的价值(也称为资产残值)，Life 为折旧期限(有时也称作资产的使用寿命)，Start_period 为进行折旧计算的起始期间，End_period 为进行折旧计算的截止期间。15

[48]. XIRR

用途：返回一组现金流的内部收益率，这些现金流不一定定期发生。若要计算一组定期现金流的内部收益率，可以使用 IRR 函数。

语法：XIRR(values, dates, guess)

参数：Values 与 dates 中的支付时间相对应的一系列现金流，Dates 是与现金流支付相对应的支付日期表，Guess 是对函数 XIRR 计算结果的估计值。

[49]. XNPV

用途：返回一组现金流的净现值，这些现金流不一定定期发生。若要计算一组定期现金流的净现值，可以使用函数 NPV。

语法：XNPV(rate, values, dates)

参数：Rate 应用于现金流的贴现率，Values 是与 dates 中的支付时间相对应的一系列现金流，Dates 与现金流支付相对应的支付日期表。

[50]. YIELD

用途：返回定期付息有价证券的收益率，函数 YIELD 用于计算债券收益率。

语法：YIELD(settlement, maturity, rate, pr, redemption, frequency, basis)

参数：Settlement 是证券的成交日，Maturity 为有价证券的到期日，Rate 为有价证券的年息票利率，Pr 为面值\$100 的有价证券的价格，Redemption 为面值\$100 的有价证券的清偿价值，Frequency 为年付息次数(如果按年支付，frequency=1；按半年期支付，frequency=2；按季支付，frequency=4)，Basis 为日计数基准类型(0 或省略为 30/360，1 为实际天数/实际天数，2 为实际天数/360，3 为实际天数/365，4 为欧洲 30/360)。

[51]. YIELDDISC

用途：返回折价发行的有价证券的年收益率。

语法：YIELDDISC(settlement, maturity, pr, redemption, basis)

参数: Settlement 为证券的成交日, Maturity 为有价证券的到期日, Pr 为面值\$100 的有价证券的价格, Redemption 为面值\$100 的有价证券的清偿价值, Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

[52]. YIELDMAT

用途: 返回到期付息的有价证券的年收益率。

语法: YIELDMAT(settlement, maturity, issue, rate, pr, basis)

参数: Settlement 是证券的成交日, Maturity 为有价证券的到期日, Issue 为有价证券的发行日(以时间序列号表示), Rate 为有价证券在发行日的利率, Pr 为面值\$100 的有价证券的价格, Basis 为日计数基准类型(0 或省略为 30/360, 1 为实际天数/实际天数, 2 为实际天数/360, 3 为实际天数/365, 4 为欧洲 30/360)。

6. 信息函数

[1]. CELL

用途: 返回某一引用区域的左上角单元格的格式、位置或内容等信息, 该函数主要用于保持与其它电子表格程序的兼容性。

语法: CELL(info_type, reference)

参数: Info_type 为一个文本值, 指明所需要的单元格信息的类型。Reference 表示要获取其有关信息的单元格。如果忽略, 则在 info_type 中所指定的信息将返回给最后更改的单元格。

实例: 公式 “=CELL("row", A20)” 返回 20, 如果 A3 单元格包含 TOTAL, 则 CELL("contents", A3)返回"TOTAL"。

[2]. ERROR.TYPE

用途: 返回对应于某一错误类型的数字, 如果没有错误则返回#N/A。在 IF 函数中可以使用 ERROR.TYPE 检测错误值, 并返回字符串(如 “发生错误”)来取代错误值。

语法: ERROR.TYPE(error_val)

参数: Error_val 为需要得到其数字代码的一个错误类型。尽管 error_val 可以是实际的错误值, 但它通常为一个单元格引用, 而此单元格中包含需要检测的公式。注意: ERROR.TYPE 函数返回的错误代码是: #NULL!返回 1, #DIV/0!返回 2, #VALUE!返回 3, #REF!返回 4, #NAME?返回 5, #NUM!返回 6, #N/A 返回 7, 其他错误则返回#N/A

实例: 如果 A5=36/0, 则公式 “=ERROR.TYPE(A5)” 返回 2。

[3]. INFO

用途: 返回有关当前操作环境的的信息。

语法: INFO(type_text)

参数: Type_text 是一文本, 指明所要返回的信息类型。其中"directory"返回当前目录或文件夹的路径, "memavail"返回可用的内存空间(以字节为单位), "memused"返回数据占用的内存空间, "numfile"返回当前所有打开的工作表的数目, 16"osversion"返回当

前操作系统的版本号(文本), "recalc"返回当前的重计算方式(“自动”或“人工”), "release"返回 Excel 的版本号(文本), "system"返回操作系统名称(Macintosh="mac", Windows="pcdos"), "totmem"返回全部内存空间, 包括已经占用的内存空间(以字节为单位)。注意: INFO 函数中的文本参数必须用西文引号引起来。

实例: 公式 “=INFO("osversion")” 返回 “Windows(32-bit)4.90”, “=INFO("memavail")” 返回 1048576。

[4]. IS 类函数

用途: 其中包括用来检验数值或引用类型的九个工作表函数。它们可以检验数值的类型并根据参数的值返回 TRUE 或 FALSE。例如, 数值为空白单元格引用时, ISBLANK 函数返回逻辑值 TRUE, 否则返回 FALSE。

语法: ISBLANK(value)、ISERR(value)、ISERROR(value)、ISLOGICAL(value)、ISNA(value)、ISNONTEXT(value)、ISNUMBER(value)、ISREF(value)、ISTEXT(value)。

参数: Value 是需要进行检验的参数。分别为空白(空白单元格)、错误值、逻辑值、文本、数字、引用值或对于以上任意参数的名称引用。如果函数中的参数为下面的内容, 则返回 TRUE: ISBLANK 的参数是空白单元格, ISERR 的参数是任意错误值(除去#N/A), ISERROR 的参数是任意错误值(#N/A、#VALUE!、#REF!、#DIV/0!、#NUM!、#NAME? 或#NULL!), ISLOGICAL 的参数是逻辑值, ISNA 的参数是错误值#N/A, ISNONTEXT 的参数是任意不是文本的内容(此函数在值为空白单元格时返回 TRUE), ISNUMBER 的参数是数字, ISREF 的参数是引用, ISTEXT 的参数是文本。注意: IS 类函数的参数 value 是不可转换的。在其他大多数需要数字的函数中, 文本"19"会被转换成数字 19。然而在公式 ISNUMBER("19")中, "19"并由文本值转换成其他类型的值, 而是返回 FALSE。IS 类函数用于检验公式计算结果十分有用, 它与函数 IF 结合在一起可以提供一种在公式中查出错误值的方法。

实例: 公式 “=ISBLANK(””)” 返回 FALSE, =ISREF(A5)返回 TRUE(其中 a5 为空白单元格)。如果需要计算 A1:A4 区域的平均值, 但不能确定单元格内是否包含数字, 则公式 AVERAGE(A1:A4) 返回错误值 #DIV/0!。为了应付这种情况, 可以使用公式 “=IF(ISERROR(AVERAGE(A1:A4))”, “引用包含空白单元格”, AVERAGE(A1:A4))” 查出可能存在的错误。

[5]. ISEVEN

用途: 测试参数的奇偶性, 如参数为偶数返回 TRUE, 否则返回 FALSE。

语法: ISEVEN(number)Number 待测试的数值。如果参数值不是整数, 则自动截去小数部分取整。注意: 该函数必须加载“分析工具库”方能使用。如果参数 number 不是数值, ISEVEN 函数返回错误值#VALUE!。

实例: 公式 “=ISEVEN(11)返回 FALSE”, =ISEVEN(6)返回 TRUE。

[6]. ISODD

用途: 测试参数的奇偶性, 如参数为奇数返回 TRUE, 否则返回 FALSE。

语法: ISODD(number)

参数: Number 待测试的数值。如参数不是整数, 则自动截去小数部分取整。注意: 该函数必须加载“分析工具库”方能使用。

实例：公式 “=ISODD(19)” 返回 TRUE，=ISODD(14.5)返回 FALSE。

[7]. N

用途：返回转化为数值后的值。

语法：N(value)

参数：Value 为要转化的值。函数 N 可以转化下表列出的值：数字返回该数字，日期返回该日期的序列号，TRUE 返回 1，FALSE 返回 0，错误值(如#DIV/0!)返回该错误值，其他值返回 0。

实例：如果 A1 包含"7"，A2 包含"文件"，A3 包含"TRUE"，则公式 “=N(A1)” 返回 7，=N(A2)返回 0(因为 A2 含有文字)，=N(A3)返回 1(因为 A3 含有 TRUE)。

[8]. NA

用途：返回错误值#N/A。#N/A 表示“无法得到有效值”。如果使用#N/A 标志空白单元格，可以避免不小心将空白单元格计算在内而产生的问题，因为公式引用含有#N/A 的单元格时返回错误值#N/A。

语法：NA()

参数：空白单元

实例：公式 “=NA(””)” 返回#N/A。17

[9]. TYPE

用途：返回数值的类型。当某一个函数的计算结果取决于特定单元格中数值的类型时，可使用函数 TYPE。

语法：TYPE(value)

参数：Value 可以是 Excel 中的数据，如数字、文本、逻辑值等等。如果 value 为数字返回 1，是文本返回 2，逻辑值返回 4，错误值返回 16，数组返回 64。

实例：如果 A1 包含文本"金额"，则公式 “=TYPE(A1)” 返回 2。=TYPE(2+A1)返回 16。

7. 逻辑运算符

[1]. AND

用途：所有参数的逻辑值为真时返回 TRUE(真)；只要有一参数的逻辑值为假，则返回 FALSE(假)。

语法：AND(logical1, logical2, .)

参数:Logical1,logical2,.为待检验的 1~30 个逻辑表达式,它们的结论或为 TRUE(真)或为 FALSE(假)。参数必须是逻辑值或者包含逻辑值的数组或引用，如果数组或引用内含文字或空白单元格，则忽略它的值。如果指定的单元格区域内包括非逻辑值，AND 将返回错误值#VALUE!。

实例：如果 A1=2、A=6，那么公式 “=AND(A1A2)” 返回 FALSE。如果 B4=104，那么公式 “=IF(AND(1<B4, B4<p>

[2]. FALSE

用途：返回逻辑值 FALSE。

语法：FALSE()

参数：该函数不需要参数。

实例：如果在 A1 单元格内输入公式 “=FALSE()”，回车后即可返回 FALSE。若在单元格或公式中输入文字 FALSE，Excel 会自动将它解释成逻辑值 FALSE。

[3]. IF

用途：执行逻辑判断，它可以根据逻辑表达式的真假，返回不同的结果，从而执行数值或公式的条件检测任务。

语法：IF(logical_test, value_if_true, value_if_false)

参数：Logical_test 计算结果为 TRUE 或 FALSE 的任何数值或表达式；Value_if_true 是 Logical_test 为 TRUE 时函数的返回值，如果 logical_test 为 TRUE 并且省略了 value_if_true，则返回 TRUE。而且 Value_if_true 可以是一个表达式；Value_if_false 是 Logical_test 为 FALSE 时函数的返回值。如果 logical_test 为 FALSE 并且省略 value_if_false，则返回 FALSE。Value_if_false 也可以是一个表达式。

实例：公式 “=IF(C2>=85,"A",IF(C2>=70,"B",IF(C2>=60,"C",IF(C2<60,"D"))))”，其中第二个 IF 语句同时也是第一个 IF 语句的参数。同样，第三个 IF 语句是第二个 IF 语句的参数，以此类推。例如，若第一个逻辑判断表达式 C2>=85 成立，则 D2 单元格被赋值“A”；如果第一个逻辑判断表达式 C2>=85 不成立，则计算第二个 IF 语句 “IF(C2>=70)”；以此类推直至计算结束，该函数广泛用于需要进行逻辑判断的场合。

[4]. NOT

用途：求出一个逻辑值或逻辑表达式的相反值。如果您要确保一个逻辑值等于其相反值，就应该使用 NOT 函数。

语法：NOT(logical)

参数：Logical 是一个可以得出 TRUE 或 FALSE 结论的逻辑值或逻辑表达式。如果逻辑值或表达式的结果为 FALSE，则 NOT 函数返回 TRUE；如果逻辑值或表达式的结果为 TRUE，那么 NOT 函数返回的结果为 FALSE。

实例：如果 A1=6、A2=8，那么公式 “=NOT(A1<A2)” 返回 TRUE。

[5]. OR

用途：所有参数中的任意一个逻辑值为真时即返回 TRUE(真)。

语法：OR(logical1, logical2, ...)

参数：Logical1, logical2, ...是需要进行检验的 1 至 30 个逻辑表达式，其结论分别为 TRUE 或 FALSE。如果数组或引用的参数包含文本、数字或空白单元格，它们将被忽略。如果指定的区域中不包含逻辑值，OR 函数将返回错误 #VALUE!。

实例：如果 A1=6、A2=8，则公式 “=OR(A1+A2>A2, A1=A2)” 返回 TRUE；而公式 “=OR(A1>A2, A1=A2)” 返回 FALSE。

[6]. TRUE

用途：返回逻辑值 TRUE。

语法：TRUE()

参数：该函数不需要参数。

实例：如果在 A1 单元格内输入公式 “=TRUE()”，回车后即可返回 TRUE。若在单元格或公式中输入文字 TRUE，Excel 会自动将它解释成逻辑值 TRUE。函数 TRUE 主要用于与其它电子表格程序兼容。

8. 查找和引用函数

[1]. ADDRESS

用途：以文字形式返回对工作簿中某一单元格的引用。

语法：ADDRESS(row_num, column_num, abs_num, a1, sheet_text)

参数：Row_num 是单元格引用中使用的行号；Column_num 是单元格引用中使用的列标；Abs_num 指明返回的引用类型(1 或省略为绝对引用，2 绝对行号、相对列标，3 相对行号、绝对列标，4 是相对引用)；A1 是一个逻辑值，它用来指明是以 A1 或 R1C1 返回引用样式。如果 A1 为 TRUE 或省略，函数 ADDRESS 返回 A1 样式的引用；如果 A1 为 FALSE，函数 ADDRESS 返回 R1C1 样式的引用。Sheet_text 为一文本，指明作为外部引用的工作表的名称，如果省略 sheet_text，则不使用任何工作表的名称

实例：公式 “=ADDRESS(1, 4, 4, 1)” 返回 D1。

[2]. AREAS

用途：返回引用中包含的区域个数。

语法：AREAS(reference)

参数：Reference 是对某一单元格或单元格区域的引用，也可以引用多个区域。注意：如果需要将几个引用指定为一个参数，则必须用括号括起来，以免 Excel 将逗号作为参数间的分隔符。

实例：公式 “=AREAS(a2:b4)” 返回 1，=AREAS((A1:A3, A4:A6, B4:B7, A16:A18)) 返回 4。

[3]. CHOOSE

用途：可以根据给定的索引值，从多达 29 个待先选参数中选出相应的值或操作。

语法：CHOOSE(index_num, value1, value2, ...)

参数：Index_num 是用来指明待选参数序号的值，它必须是 1 到 29 之间的数字、或者是包含数字 1 到 29 的公式或单元格引用；Value1, value2, ... 为 1 到 29 个数值参数，可以是数字、单元格，已定义的名称、公式、函数或文本。

实例：公式 “=CHOOSE(2, “电脑”, “爱好者”) 返回 “爱好者”。公式 “=SUM(A1:CHOOSE(3, A10, A20, A30))” 与公式 “=SUM(A1:A30)” 等价(因为 CHOOSE(3, A10, A20, A30) 返回 A30)。

[4]. COLUMN

用途：返回给定引用的列标。

语法：COLUMN(reference)

参数：Reference 为需要得到其列标的单元格或单元格区域。如果省略 reference，则假定函数 COLUMN 是对所在单元格的引用。如果 reference 为一个单元格区域，并且函数 COLUMN 作为水平数组输入，则 COLUMN 函数将 reference 中的列标以水平数组的形式返回。

实例：公式 “=COLUMN(A3)” 返回 1，=COLUMN(B3:C5)返回 2。

[5]. COLUMNS

用途：返回数组或引用的列数。

语法：COLUMNS(array)

参数：Array 为需要得到其列数的数组、数组公式或对单元格区域的引用。

实例：公式 “=COLUMNS(B1:C4)” 返回 2，=COLUMNS({5, 4; 4, 5})返回 2。

[6]. HLOOKUP

用途：在表格或数值数组的首行查找指定的数值，并由此返回表格或数组当前列中指定行处的数值。

语法：HLOOKUP(lookup_value, table_array, row_index_num, range_lookup)

参数：Lookup_value 是需要在数据表第一行中查找的数值，它可以是数值、引用或文字串；Table_array 是需要在其中查找数据的数据表，可以使用对区域或区域名称的引用，Table_array 的第一行的数值可以是文本、数字或逻辑值。Row_index_num 为 table_array 中待返回的匹配值的行序号。Range_lookup 为一逻辑值，指明函数 HLOOKUP 查找时是精确匹配，还是近似匹配

实例：如果 A1:B3 区域存放的数据为 34、23、68、69、92、36，则公式 “=HLOOKUP(34, A1:B3, 1, FALSE)返回 34；=HLOOKUP(3, {1, 2, 3;“a”, “b”, “c”;“d”, “e”, “f”}, 2, TRUE)返回 “c”。

[7]. HYPERLINK

用途：创建一个快捷方式，用以打开存储在网络服务器、Intranet(Internet)或本地硬盘的其它文件。

语法：HYPERLINK(link_location, friendly_name)

参数：Link_location 是文件的路径和文件名，它还可以指向文档中的某个更为具体的位置，如 Excel 工作表或工作簿中特定的单元格或命名区域，或是指向 Word 文档中的书签。路径可以是存储在硬盘驱动器上的文件，或是 Internet 或 Intranet 上的 URL 路径；Friendly_name 为单元格中显示的链接文字或数字，它用蓝色显示并带有下划线。如果省略了 Friendly_name，单元格就将 link_location 显示为链接。

实例：HYPERLINK(“http://www.mydrivers.com/”, “驱动之家”)会在工作表中显示文本 “驱动之家”，单击它即可连接到 “http://www.mydrivers.com/”。公式 “=HYPERLINK(“D:\README.TXT”, “说明文件”)” 在工作表中建立一个的蓝色 “说明文件” 链接，单击它可以打开 D 盘上的 README.TXT 文件。

[8]. INDEX

用途：返回表格或区域中的数值或对数值的引用。函数 INDEX() 有两种形式：数组和引用。数组形式通常返回数值或数值数组；引用形式通常返回引用。

语法：INDEX(array, row_num, column_num) 返回数组中指定的单元格或单元格数组的数值。INDEX(reference, row_num, column_num, area_num) 返回引用中指定单元格或单元格区域的引用)。

参数：Array 为单元格区域或数组常数；Row_num 为数组中某行的行序号，函数从该行返回数值。如果省略 row_num，则必须有 column_num；Column_num 是数组中某列的列序号，函数从该列返回数值。如果省略 column_num，则必须有 row_num。Reference 是对一个或多个单元格区域的引用，如果为引用输入一个不连续的选定区域，必须用括号括起来。Area_num 是选择引用中的一个区域，并返回该区域中 row_num 和 column_num 的交叉区域。选中或输入的的第一个区域序号为 1，第二个为 2，以此类推。如果省略 area_num，则 INDEX 函数使用区域 1。

实例：如果 A1=68、A2=96、A3=90，则公式 “=INDEX(A1:A3, 1, 1)” 返回 68，=INDEX(A1:A3, 1, 1, 1) 返回 68。

[9]. INDIRECT

用途：返回由文字串指定的引用。此函数立即对引用进行计算，并显示其内容。当需要更改公式中单元格的引用，而不更改公式本身，即可使用 INDIRECT 函数。

语法：INDIRECT(ref_text, a1)

参数：Ref_text 是对单元格的引用，此单元格可以包含 A1 样式的引用、R1C1 样式的引用、定义为引用的名称或对文字串单元格的引用；A1 为一逻辑值，指明包含在单元格 ref_text 中的引用的类型。如果 a1 为 TRUE 或省略，ref_text 被解释为 A1-样式的引用。如果 a1 为 FALSE，ref_text 被解释为 R1C1-样式的引用。

实例：如果单元格 A1 存放有文本 B1，而 B1 单元格中存放了数值 68.75，则公式 “=INDIRECT(\$A\$1)” 返回 68.75。

[10]. LOOKUP

用途：返回向量(单行区域或单列区域)或数组中的数值。该函数有两种语法形式：向量和数组，其向量形式是在单行区域或单列区域(向量)中查找数值，然后返回第二个单行区域或单列区域中相同位置的数值；其数组形式在数组的第一行或第一列查找指定的数值，然后返回数组的最后一行或最后一列中相同位置的数值。

语法 1(向量形式)：LOOKUP(lookup_value, lookup_vector, result_vector)

语法 2(数组形式)：LOOKUP(lookup_value, array)

参数 1(向量形式)：Lookup_value 为函数 LOOKUP 在第一个向量中所要查找的数值。Lookup_value 可以为数字、文本、逻辑值或包含数值的名称或引用。Lookup_vector 为只包含一行或一列的区域。Lookup_vector 的数值可以为文本、数字或逻辑值。

参数 2(数组形式)：Lookup_value 为函数 LOOKUP 在数组中所要查找的数值。Lookup_value 可以为数字、文本、逻辑值或包含数值的名称或引用。如果函数 LOOKUP 找不到 lookup_value，则使用数组中小于或等于 lookup_value 的最大数值。Array 为包含文本、数字或逻辑值的单元格区域，它的值用于与 lookup_value 进行比较。

注意：Lookup_vector 的数值必须按升序排列，否则 LOOKUP 函数不能返回正确的

结果，参数中的文本不区分大小写。

实例：如果 A1=68、A2=76、A3=85、A4=90，则公式 “=LOOKUP(76, A1:A4)” 返回 2，=LOOKUP("bump", {"a", 1;"b", 2;"c", 3})返回 2。

[11]. MATCH

用途：返回在指定方式下与指定数值匹配的数组中元素的相应位置。如果需要找出匹配元素的位置而不是匹配元素本身，则应该使用 MATCH 函数。

语法：MATCH(lookup_value, lookup_array, match_type)

参数：Lookup_value 为需要在数据表中查找的数值，它可以是数值(或数字、文本或逻辑值)、对数字、文本或逻辑值的单元格引用。Lookup_array 是可能包含所要查找的数值的连续单元格区域，Lookup_array 可以是数组或数组引用；Match_type 为数字-1、0 或 1，它说明 Excel 如何在 lookup_array 中查找 lookup_value。如果 match_type 为 1，函数 MATCH 查找小于或等于 lookup_value 的最大数值。如果 match_type 为 0，函数 MATCH 查找等于 lookup_value 的第一个数值。如果 match_type 为-1，函数 MATCH 查找大于或等于 lookup_value 的最小数值。注意：MATCH 函数返回 lookup_array 中目标值的位置，而不是数值本身。如果 match_type 为 0 且 lookup_value 为文本，lookup_value 可以包含通配符(“*”和“?”)。星号可以匹配任何字符序列，问号可以匹配单个字符。

实例：如果 A1=68、A2=76、A3=85、A4=90，则公式 “=MATCH(90, A1:A5, 0)” 返回 3。

[12]. OFFSET

用途：以指定的引用为参照系，通过给定偏移量得到新的引用。返回的引用可以是一个单元格或单元格区域，并可以指定返回的行数或列数

语法：OFFSET(reference, rows, cols, height, width)

参数：Reference 是作为偏移量参照系的引用区域，它必须是单元格或相连单元格区域的引用；Rows 是相对于偏移量参照系的左上角单元格，上(下)偏移的行数。如果使用 5 作为参数 Rows，则说明目标引用区域的左上角单元格比 reference 低 5 行。行数可为正数(代表在起始引用的下方)或负数(代表在起始引用的上方)；Cols 是相对于偏移量参照系的左上角单元格，左(右)偏移的列数。如果使用 5 作为参数 Cols，则说明目标引用区域的左上角的单元格比 reference 靠右 5 列。列数可为正数(代表在起始引用的右边)或负数(代表在起始引用的左边)；Height 是要返回的引用区域的行数，Height 必须为正数；Width 是要返回的引用区域的列数，Width 必须为正数。

实例：如果 A1=68、A2=76、A3=85、A4=90，则公式 “=SUM(OFFSET(A1:A2, 2, 0, 2, 1))” 返回 177。

[13]. ROW

用途：返回给定引用的行号

语法：ROW(reference)。Reference 为需要得到其行号的单元格或单元格区域

实例：公式 “=ROW(A6)” 返回 6，如果在 C5 单元格中输入公式 “=ROW()”，其计算结果为 5。

[14]. ROWS

用途：返回引用或数组的行数

语法：ROWS(array)

参数：Array 是需要得到其行数的数组、数组公式或对单元格区域的引用

实例：公式 “=ROWS(A1:A9)” 返回 9，=ROWS({1, 2, 3; 4, 5, 6; 1, 2, 3}) 返回 3。

[15]. RTD

用途：从支持 COM 自动化的程序中返回实时数据

语法：RTD(ProgID, server, topic1, [topic2], ...)

参数：ProgID 已安装在本地计算机中，经过注册的 COM 自动化加载宏的 ProgID 名称，该名称用引号引起来。Server 是运行加载宏的服务器的名称。如果没有服务器，程序是在本地计算机上运行，那么该参数为空白。topic1, topic2, ...21 为 1 到 28 个参数，这些参数放在一起代表一个唯一的实时数据。

[16]. TRANSPOSE

用途：返回区域的转置(所谓转置就是将数组的第一行作为新数组的第一列，数组的第二行作为新数组的第二列，以此类推)。

语法：TRANSPOSE(array)

参数：Array 是需要转置的数组或工作表中的单元格区域。

实例：如果 A1=68、A2=76、B1=85、B2=90，那么公式 “={TRANSPOSE(A1:B1)}” 返回 C1=56、D1=98、C2=90、D2=87。

[17]. VLOOKUP

用途：在表格或数值数组的首列查找指定的数值，并由此返回表格或数组当前行中指定列处的数值。当比较值位于数据表首列时，可以使用函数 VLOOKUP 代替函数 HLOOKUP。

语法：VLOOKUP(lookup_value, table_array, col_index_num, range_lookup)

参数：Lookup_value 为需要在数据表第一列中查找的数值，它可以是数值、引用或文字串。Table_array 为需要在其中查找数据的数据表，可以使用对区域或区域名称的引用。Col_index_num 为 table_array 中待返回的匹配值的列序号。Col_index_num 为 1 时，返回 table_array 第一列中的数值；col_index_num 为 2，返回 table_array 第二列中的数值，以此类推。Range_lookup 为一逻辑值，指明函数 VLOOKUP 返回时是精确匹配还是近似匹配。如果为 TRUE 或省略，则返回近似匹配值，也就是说，如果找不到精确匹配值，则返回小于 lookup_value 的最大数值；如果 range_value 为 FALSE，函数 VLOOKUP 将返回精确匹配值。如果找不到，则返回错误值 #N/A。

实例：如果 A1=23、A2=45、A3=50、A4=65，则公式 “=VLOOKUP(50, A1:A4, 1, TRUE)” 返回 50。

9. 数学和三角函数

[1]. ABS

用途：返回某一参数的绝对值。

语法：ABS(number)

参数：number 是需要计算其绝对值的一个实数。

实例：如果 A1=-16，则公式 “=ABS(A1)” 返回 16。

[2]. ACOS

用途：返回以弧度表示参数的反余弦值，范围是 $0 \sim \pi$ 。

语法：ACOS(number)

参数：number 是某一角度的余弦值，大小在 $-1 \sim 1$ 之间。

实例：如果 A1=0.5，则公式 “=ACOS(A1)” 返回 1.047197551(即 $\pi/3$ 弧度，也就是 60°)；而公式 “=ACOS(-0.5)*180/PI()” 返回 120°。

[3]. ACOSH

用途：返回参数的反双曲余弦值。

语法：ACOSH(number)

参数：number 必须大于或等于 1。

实例：公式 “=ACOSH(1)” 的计算结果等于 0；“=ACOSH(10)” 的计算结果等于 2.993223。

[4]. ASIN

用途：返回参数的反正弦值。

语法：ASIN(number)

参数：Number 为某一角度的正弦值，其大小介于 $-1 \sim 1$ 之间。

实例：如果 A1=-0.5，则公式 “=ASIN(A1)” 返回 -0.5236($-\pi/6$ 弧度)；而公式 “=ASIN(A1)*180/PI()” 返回 -30°。

[5]. ASINH

用途：返回参数的反双曲正弦值。

语法：ASINH(number)

参数：number 为任意实数。

实例：公式 “=ASINH(-2.5)” 返回 -1.64723；“=ASINH(10)” 返回 2.998223。

[6]. ATAN

用途：返回参数的反正切值。返回的数值以弧度表示，大小在 $-\pi/2 \sim \pi/2$ 之间。

语法：ATAN(number)

参数：number 为某一角度的正切值。如果要用度表示返回的反正切值，需将结果乘

以 $180/\text{PI}()$ 。

实例：公式 “=ATAN(1)” 返回 $0.785398(\pi/4 \text{ 弧度})$ ；=ATAN(1)*180/PI() 返回 450。

[7]. ATAN2

用途：返回直角坐标系中给定 X 及 Y 的反正切值。它等于 X 轴与过原点和给定点 (x_num, y_num) 的直线之间的夹角，并介于 $-\pi \sim \pi$ 之间(以弧度表示，不包括 $-\pi$)。

语法：ATAN2(x_num, y_num)

参数：X_num 为给定点的 X 坐标，Y_num 为给定点的 Y 坐标。

实例：公式 “=ATAN2(1, 1)” 返回 $0.785398(\text{即 } \pi/4 \text{ 弧度})$ ；=ATAN2(-1, -1) 返回 $-2.35619(-3\pi/4 \text{ 弧度})$ ；=ATAN2(-1, -1)*180/PI() 返回 -1350。

[8]. ATANH

用途：返参数的反双曲正切值。参数必须在 $-1 \sim 1$ 之间(不包括 -1 和 1)。

语法：ATANH(number)

参数：number 是 $-1 < \text{NUMBER} < 1$ 的任意实数。

实例：公式 “=ATANH(0.5)” 返回 0.549306144；=ATANH(-0.1) 返回 -0.10034。

[9]. CEILING

用途：将参数 Number 沿绝对值增大的方向，返回一个最接近的整数或基数 significance 的最小倍数。

语法：CEILING(number, significance)

参数：number 为待返回的数值，Significance 为待返回的最小倍数。注意：无论 number 的正负如何，都是按远离 0 点的方向返回结果。如果 number 是 Significance 的倍数，则返回的数值是其自身。

实例：如果 A1=3.1416，则公式 “=CEILING(A1, 1)” 返回的结果是 4；=CEILING(-2.5, -2) 返回的结果为 -4。

[10]. COMBIN

用途：返回一组对象所有可能的组合数目。

语法：COMBIN(number, number_chosen)

参数：number 是某一对象的总数量，number_chosen 则是每一组合中对象的数量。

注意：函数中的参数按照截尾取整的原则参与运算，并且要求 $\text{number} > 0$ 、 $\text{number_chosen} > 0$ 以及 $\text{number} > \text{number_chosen}$ 。

实例：假设有 10 名乒乓球队员，从中选出任意两人搭配参加双打，则计算公式为 “=COMBIN(10, 2)”，可以得出 45 种搭配方案。

[11]. COS

用途：返回某一角度的余弦值。

语法：COS(number)

参数：number 为需要余弦值的一个角度，必须用弧度表示。如果 number 的单位是度，可以乘以 $\text{PI}()/180$ 转换为弧度。

实例：如果 A1=1，则公式 “=COS(A1)” 返回 0.540302；若 A2=60，则公式 “=COS(A2*PI()/180)” 返回 0.5。

[12]. COSH

用途：返回参数的双曲余弦值。

语法：COSH(number)

参数：number 为任意实数。

实例：如果 A1=5、A3=6，则公式 “=COSH(A1+A3)” 返回 29937.07087；若 C1=60，则公式 “=COSH(COS(C1*PI()/180))” 返回 1.127625965。

[13]. COUNTIF

用途：统计某一区域中符合条件的单元格数目。

语法：COUNTIF(range, criteria)

参数：range 为需要统计的符合条件的单元格数目的区域；Criteria 为参与计算的单元条件，其形式可以为数字、表达式或文本(如 36、“>160”和“男”等)。其中数字可以直接写入，表达式和文本必须加引号。

实例：假设 A1:A5 区域内存放的文本分别为女、男、女、男、女，则公式 “=COUNTIF(A1:A5, “女”)” 返回 3。

[14]. DEGREES

用途：将弧度转换为度。

语法：DEGREES(angle)

参数：angle 是采用弧度单位的一个角度。

实例：公式 “=DEGREES(1)” 返回 57.29577951”，=DEGREES(PI()/3) 返回 60。

[15]. EVEN

用途：返回沿绝对值增大方向，将一个数值取整为最接近的偶数。

语法：EVEN(number)

参数：number 是要取整的一个数值。

实例：如果 A1=-2.6 则公式 “=EVEN(A1)” 返回 -4；=EVEN(-4.56+6.87) 返回 4。

[16]. EXP

用途：返回 e 的 n 次幂。

语法：EXP(number)

参数：Number 为底数 e 的指数。注意：EXP 函数是计算自然对数的 LN 函数的反函数。

实例：如果 A1=3，则公式 “=EXP(A1)” 返回 20.085537 即 e³。

[17]. FACT

用途：返回一个数的阶乘，即 1*2*3*…*该数。

语法: FACT(number)注意: Number 是计算其阶乘的非负数。如果输入的 Number 不是整数, 则截去小数部分取整数。

实例: 如果 A1=3, 则公式 “=FACT(A1)” 返回 6; =FACT(5.5)返回 1*2*3*4*5.5 即 120。

[18]. FACTDOUBLE

用途: 返回参数 Number 的半阶乘。

语法: FACTDOUBLE(number)Number 要计算其半阶乘的数值, 如

参数 Number 为非整数, 则截尾取整。注意: 如果该函数不存在, 应当运行 “安装” 程序加载 “分析工具库”

实例: 公式 “=FACTDOUBLE(4)” 返回 8。

[19]. FLOOR

用途:

参数 Number 沿绝对值减小的方向去尾舍入, 使其等于最接近的 significance 的倍数

语法: FLOOR(number, significance)

参数: Number 为要舍入的某一数值, Significance 为该数值的倍数

实例: 如果 A1=22.5, 则公式 “=FLOOR(A1, 1)” 返回 22; =FLOOR(-2.5, -2)返回 -2。

[20]. GCD

用途: 返回两个或多个整数的最大公约数。

语法: GCD(number1, number2, ...)

参数: Number1, number2, ...为 1 到 29 个数值, 如果数值为非整数, 则截尾取整。

说明: 如果该函数不存在, 必须运行 “安装” 程序加载 “分析工具库”

实例: 如果 A1=16、A2=28、A3=46, 则公式 “=GCD(A1: A3)” 返回 2。

[21]. INT

用途: 将任意实数向下取整为最接近的整数。

语法: INT(number)

参数: Number 为需要处理的任意一个实数。

实例: 如果 A1=16.24、A2=-28.389, 则公式 “=INT(A1)” 返回 16, =INT(A2)返回-29。

[22]. LCM

用途: 返回整数的最小公倍数。最小公倍数是所有整数参数 number1、number2、.. 的最小正整数倍数。用函数 LCM 可以将分母不同的分数相加。

语法: LCM(number1, number2, ...)

参数: Number1, number2, ...是要计算最小公倍数的 1 到 29 个参数。如果参数不是整数, 则自动截去小数部分取整。说明: 该函数需要加载 “分析工具库” 才能使用

实例: 如果 A1=4、A2=16、A3=8, 则公式 “=LCM(A1:A3)” 返回 16。

[23]. LN

用途: 返回一个数的自然对数, 即以 $e(2.71828182845904)$ 为底的对数(LN 函数是 EXP 函数的反函数)。

语法: LN(number)

参数: Number 是待计算其自然对数的正实数。

实例: 如果 A1=100、A2=67, 则公式 “=LN(A1+A2)” 返回 5.117993812; =LN(EXP(3)) 返回 3; =EXP(LN(4)) 返回 4。

[24]. LOG

用途: 按所指定的底数, 返回某个数的对数。

语法: LOG(number, base)

参数: Number 是计算对数的任意实数, Base 是对数的底数。如果省略底数, 则默认它的值为 10。

实例: 如果 A1=8, 则公式 “=LOG(A1, 2)” 返回 3; =LOG(100, 10) 返回 2。

[25]. LOG10

用途: 返回以 10 为底的对数。

语法: LOG10(number)

参数: Number 是待计算常用对数的一个正实数。

实例: 如果 A1=1000, 则公式 “=LOG10(A1)” 返回 3; =LOG10(10^5) 返回 5。

[26]. MDETERM

用途: 返回一个数组的矩阵行列式的值。

语法: MDETERM(array)

参数: Array 是一个行列数相等的数值数组。Array 可以是单元格区域, 例如 A1:C3; 或是一个数组常量, 如 {1, 2, 3; 4, 5, 6; 7, 8, 9}; 也可以是区域或数组常量的名称。矩阵行列式的值多用于求解多元联立方程。

实例: 如果 A1=1、A2=2、B1=3、B2=4, 则公式 “=MDETERM(A1:B2)” 返回 -2。

[27]. MINVERSE

用途: 返回数组矩阵的逆矩阵。

语法: MINVERSE(array)

参数: Array 是具有相等行列数的数值数组, 它可以是单元格区域, 例如 A1:C3; 也可以是常数数组如 {1, 2, 3; 4, 5, 6; 7, 8, 9}; 或者是两者的名称。

实例: 公式 “=MINVERSE({4, -1; 2, 0})” 返回 {0, 0.5; -1, 2}; =MINVERSE({1, 2, 1; 3, 4, -1; 0, 2, 0}) 返回 {0.25, 0.25, -0.75; 0, 0, 0.5; 0.75, -0.25, -0.25}。

[28]. MMULT

用途: 返回两数组的矩阵乘积。结果矩阵的行数与 array1 的行数相同, 矩阵的列数与 array2 的列数相同。

语法: MMULT(array1, array2)

参数: Array1 和 array2 是要进行矩阵乘法运算的两个数组。Array1 的列数必须与 array2 的行数相同, 而且两个数组中都只能包含数值。Array1 和 array2 可以是单元格区域、数组常数或引用。

实例: 公式 “=MMULT({1, 2;2, 3}, {3, 4;4, 5})” 返回 11。

[29]. MOD

用途: 返回两数相除的余数, 其结果的正负号与除数相同。

语法: MOD(number, divisor)

参数: Number 为被除数, Divisor 为除数(divisor 不能为零)。

实例: 如果 A1=51, 则公式 “=MOD(A1, 4)” 返回 3; =MOD(-101, -2)返回 -1。

[30]. MROUND

用途: 返回参数按指定基数舍入后的数值。

语法: MROUND(number, significance)

参数: Number 是要舍入的数值, Significance 是要对参数 Number 进行舍入运算的基数。注意: 如果参数 number 除以基数 Significance 的余数大于或等于基数 Significance 的一半, 则函数 MROUND 向远离零的方向舍入。另外, 该函数只有加载了“分析工具库”方可使用。

实例: 如果 A1=6.6876, 则公式 “=MROUND(A1, 4)” 的计算结果是 8。

[31]. MULTINOMIAL

用途: 返回参数和的阶乘与。参数阶乘乘积的比值, 例如 MULTINOMIAL(2, 3, 4) 执行的运算为 $9! / 2! * 3! * 4!$ 。

语法: MULTINOMIAL(number1, number2, ...)

参数: Number1, number2, ...是用于进行函数 Multinomial 运算的 1 到 29 个数值参数。注意: 该函数只有加载“分析工具库”方可使用。

实例: MULTINOMIAL(2, 3, 4)返回的结果为 1260。

[32]. ODD

用途: 将一个正(负数)向上(向下)舍入为最接近的奇数。

语法: ODD(number)

参数: Number 是待计算的一个数值。注意: 参数 number 必须是一个数值参数, 不论它的正负号如何, 其结果均按远离 0 的方向舍入。如果 number 恰好是奇数, 则保持原来的数值不变。

实例: 如果 A1=31.5, 则公式 “=ODD(A1)” 返回 33; =ODD(3)返回 3; =ODD(-26.38)返回 -27。

[33]. PI

用途: 返回圆周率 π , 精确到小数点后 14 位。

语法: PI()

参数：不需要参数。

实例：公式 “=PI()” 返回 3.14159265358979。

[34]. POWER

用途：返回给定数字的乘幂。

语法：POWER(number, power)

参数：其中 Number 为底数，Power 为指数，均可以为任意实数。注意：可以用 “^” 运算符代替 POWER 函数执行乘幂运算，例如公式 “=5^2” 与 “=POWER(5, 2)” 等价

实例：如果 A1=25.37, 则公式 “=POWER(A1, 7)” 返回 6764617901; =POWER(4, 5/4)返回 5.656854。

[35]. PRODUCT

用途：将所有数字形式给出的参数相乘，然后返回乘积值。

语法：PRODUCT(number1, number2, ...)

参数：Number1, number2, ...为 1 到 30 个需要相乘的数字参数。

实例：如果单元格 A1=24、A2=36、A3=80, 则公式 “=PRODUCT(A1:A3)” 返回 69120; =PRODUCT(12, 26, 39)返回 12168。

[36]. QUOTIENT

用途：返回商的整数部分，即舍去商的小数部分。

语法：QUOTIENT(numerator, denominator)

参数：Numerator 为被除数，Denominator 为除数。注意：该函数只有加载 “分析工具库” 方可使用。

实例：如果 A1=86、A2=9, 则公式 “=QUOTIENT(A1, A2)” 返回 9; =QUOTIENT(-10, 3)返回 -3。

[37]. RADIANS

用途：将一个表示角度的数值或参数转换为弧度。

语法：RADIANS(angle)

参数：Angle 为需要转换成弧度的角度。

实例：如果 A1=90, 则公式 “=RADIANS(A1)” 返回 1.57, =RADIANS(360)返回 6.28(均取两位小数)。

[38]. RAND

用途：返回一个大于等于 0 小于 1 的随机数，每次计算工作表(按 F9 键)将返回一个新的数值。

语法：RAND()

参数：不需要注意：如果要生成 a, b 之间的随机实数，可以使用公式 “=RAND()*(b-a)+a”。如果在某一单元格内应用公式 “=RAND()”，然后在编辑状态下按住 F9 键，将会产生一个变化的随机数。

实例：公式 “=RAND()*1000” 返回一个大于等于 0、小于 1000 的随机数。

[39]. RANDBETWEEN

用途：产生位于两个指定数值之间的一个随机数，每次重新计算工作表(按 F9 键)都将返回新的数值。

语法：RANDBETWEEN(bottom, top)

参数：Bottom 是 RANDBETWEEN 函数可能返回的最小随机数，Top 是 RANDBETWEEN 函数可能返回的最大随机数。注意：该函数只有在加载了“分析工具库”以后才能使用。

实例：公式“=RANDBETWEEN(1000, 9999)”将返回一个大于等于 1000、小于等于 9999 的随机数。

[40]. ROMAN

用途：将阿拉伯数字转换为文本形式的罗马数字。

语法：ROMAN(number, form)

参数：Number 为需要转换的阿拉伯数字。Form 则是一个数字，它指定要转换的罗马数字样式。可以从经典到简化，随着 form 值的增加趋于简单。

实例：公式“=ROMAN(499, 0)”返回“CDXCIX”；=ROMAN(499, 1)返回“LDV LIV”。

[41]. ROUND

用途：按指定位数四舍五入某个数字。

语法：ROUND(number, num_digits)

参数：Number 是需要四舍五入的数字；Num_digits 为指定的位数，Number 按此位数进行处理。注意：如果 num_digits 大于 0，则四舍五入到指定的小数位；如果 num_digits 等于 0，则四舍五入到最接近的整数；如果 num_digits 小于 0，则在小数点左侧按指定位数四舍五入。

实例：如果 A1=65.25，则公式“=ROUND(A1, 1)”返回 65.3；=ROUND(82.149, 2)返回 82.15；=ROUND(21.5, -1)返回 20。

[42]. ROUNDOWN

用途：按绝对值减小的方向舍入某一数字。

语法：ROUNDOWN(number, num_digits)

参数：Number 是需要向下舍入的任意实数，Num_digits 指定计算的小数位数。注意：ROUNDOWN 函数和 ROUND 函数的用途相似，不同之处是 ROUNDOWN 函数总是向下舍入数字。

实例：如果 A1=65.251，则公式“=ROUNDOWN(A1, 0)”返回 65；=ROUNDOWN(A1, 2)返回 65.25；=ROUNDOWN(3.14159, 3)返回 3.141；=ROUNDOWN(-3.14159, 1)返回-3.1；=ROUNDOWN(31415.92654, -2)返回 31400。

[43]. ROUNDUP

用途：按绝对值增大的方向舍入一个数字。

语法: ROUNDUP(number, num_digits)

参数: Number 为需要舍入的任意实数, Num_digits 指定舍入的数字位数。注意: 如果 num_digits 为 0 或省略, 则将数字向上舍入到最接近的整数。如果 num_digits 小于 0, 则将数字向上舍入到小数点左边的相应位数。

实例: 如果 A1=65.251, 则公式 “=ROUNDUP(A1, 0)” 返回 66; =ROUNDUP(A1, 1) 返回 66; =ROUNDUP(A1, 2) 返回 65.26; =ROUNDUP(-3.14159, 1) 返回 -3.2; =ROUNDUP(31415.92654, -2) 返回 31500。

[44]. SERIESSUM

用途: 返回幂级数的和。

语法: SERIESSUM(x, n, m, coefficients)

参数: X 幂级数的输入值, N 为 x 的首项乘幂, M 为级数中每一项的乘幂 n 的步长增加值, Coefficients 为一系列与 x 各级乘幂相乘的系数。Coefficients 的值决定了幂级数的项数。注意: SERIESSUM 函数只有加载 “分析工具库” 以后方能使用。

实例: 如果单元格 A1=65.25, 则公式 “=SERIESSUM(A1, 3, 2, 6)” 返回 1666835.719。

[45]. SIGN

用途: 返回数字的符号。正数返回 1, 零返回 0, 负数时返回 -1。

语法: SIGN(number)

参数: Number 是需要返回符号的任意实数。

实例: 如果 A1=65.25, 则公式 “=SIGN(A1)” 返回 1; =SIGN(6-12) 返回 -1; =SIGN(9-9) 返回 0。

[46]. SIN

用途: 返回某一角度的正弦值。

语法: SIN(number)

参数: Number 是待求正弦值的一个角度(采用弧度单位), 如果它的单位是度, 则必须乘以 PI()/180 转换为弧度。

实例: 如果 A1=60, 则公式 “=SIN(A1*PI()/180)” 返回 0.866, 即 60 度角的正弦值。

[47]. SINH

用途: 返回任意实数的双曲正弦值。

语法: SINH(number)

参数: Number 为任意实数。

实例: 公式 “=SINH(10)” 返回 11013.23287, =SINH(-6) 返回 -201.7131574。

[48]. SQRT

用途: 返回某一正数的算术平方根。

语法: SQRT(number)

参数: Number 为需要求平方根的一个正数。

实例: 如果 A1=81, 则公式 “=SQRT(A1)” 返回 9; =SQRT(4+12) 返回 6。

[49]. SQRTPI

用途：返回一个正实数与 π 的乘积的平方根。

语法：SQRTPI(number)

参数：Number 是用来与 π 相乘的正实数。注意：SQRTPI 函数只有加载“分析工具库”以后方能使用。如果参数 number<0，则函数 SQRTPI 返回错误值#NUM!。

实例：公式“=SQRTPI(1)”返回 1.772454，=SQRTPI(2)返回 2.506628。

[50]. SUBTOTAL

用途：返回数据清单或数据库中的分类汇总。如果用户使用“数据”菜单中的“分类汇总”命令创建了分类汇总数据清单，即可编辑 SUBTOTAL 函数对其进行修改。

语法：SUBTOTAL(function_num, ref1, ref2.)

参数：Function_num 为 1 到 11 之间的自然数，用来指定分类汇总计算使用的函数(1 是 AVERAGE; 2 是 COUNT; 3 是 COUNTA; 4 是 MAX; 5 是 MIN; 6 是 PRODUCT; 7 是 STDEV; 8 是 STDEVP; 9 是 SUM; 10 是 VAR; 11 是 VARP)。Ref1、ref2.则是需要分类汇总的 1 到 29 个区域或引用。

实例：如果 A1=1、A2=2、A3=3，则公式“=SUBTOTAL(9, A1:A3)”将使用 SUM 函数对“A1:A3”区域进行分类汇总，其结果为 6。

[51]. SUM

用途：返回某一单元格区域中所有数字之和。

语法：SUM(number1, number2, ...)

参数：Number1, number2, ...为 1 到 30 个需要求和的数值(包括逻辑值及文本表达式)、区域或引用。注意：参数表中的数字、逻辑值及数字的文本表达式可以参与计算，其中逻辑值被转换为 1、文本被转换为数字。如果参数为数组或引用，只有其中的数字将被计算，数组或引用中的空白单元格、逻辑值、文本或错误值将被忽略。

实例：如果 A1=1、A2=2、A3=3，则公式“=SUM(A1: A3)”返回 6；=SUM("3", 2, TRUE)返回 6，因为"3"被转换成数字 3，而逻辑值 TRUE 被转换成数字 1。

[52]. SUMIF

用途：根据指定条件对若干单元格、区域或引用求和。

语法：SUMIF(range, criteria, sum_range)

参数：Range 为用于条件判断的单元格区域，Criteria 是由数字、逻辑表达式等组成的判定条件，Sum_range 为需要求和的单元格、区域或引用。

实例：某单位统计工资报表中职称为“中级”的员工工资总额。假设工资总额存放在工作表的 F 列，员工职称存放在工作表 B 列。则公式为“=SUMIF(B1:B1000, "中级", F1:F1000)”，其中“B1:B1000”为提供逻辑判断依据的单元格区域，“中级”为判断条件，就是仅仅统计 B1:B1000 区域中职称为“中级”的单元格，F1:F1000 为实际求和的单元格区域。

[53]. SUMPRODUCT

用途：在给定的几组数组中，将数组间对应的元素相乘，并返回乘积之和。

语法：SUMPRODUCT(array1, array2, array3, ...)

参数：Array1, array2, array3, ...为 2 至 30 个数组，其相应元素需要进行相乘并求和。

实例：公式 “=SUMPRODUCT({3, 4;8, 6;1, 9}, {2, 7;6, 7;5, 3})” 的计算结果是 156。

[54]. SUMSQ

用途：返回所有参数的平方和。

语法：SUMSQ(number1, number2, ...)

参数：Number1, number2, ...为 1 到 30 个需要求平方和的参数，它可以是数值、区域、引用或数组。

实例：如果 A1=1、A2=2、A3=3, 则公式 “=SUMSQ(A1:A3) 返回 14 (即 1²+2²+3²=14)。

[55]. SUMX2MY2

用途：返回两数组中对应数值的平方差之和。

语法：SUMX2MY2(array_x, array_y)

参数：Array_x 为第一个数组或数值区域。Array_y 为第二个数组或数值区域。

实例：如果 A1=1、A2=2、A3=3、B1=4、B2=5、B3=6, 则公式 “=SUMX2MY2(A1:A3, B1:B3)” 返回 -63。

[56]. SUMX2PY2

用途：返回两数组中对应数值的平方和的总和，此类运算在统计中经常遇到。

语法：SUMX2PY2(array_x, array_y)

参数：Array_x 为第一个数组或数值区域，Array_y 为第二个数组或数值区域。

实例：如果 A1=1、A2=2、A3=3、B1=4、B2=5、B3=6, 则公式 “=SUMX2PY2(A1:A3, B1:B3)” 返回 91。

[57]. SUMXMY2

用途：返回两数组中对应数值之差的平方和。

语法：SUMXMY2(array_x, array_y)

参数：Array_x 为第一个数组或数值区域。Array_y 为第二个数组或数值区域。

实例：如果 A1=1、A2=2、A3=3、B1=4、B2=5、B3=6, 则公式 “=SUMXMY2(A1:A3, B1:B3)” 返回 27。

[58]. TAN

用途：返回某一角度的正切值。

语法：TAN(number)

参数：Number 为要求正切的角度，以弧度表示。如果该参数的单位是度，可以乘

以 $\text{PI}()/180$ 转换为弧度。

实例：如果 $A1=60$ ，则公式 “ $=\text{TAN}(A1*\text{PI}()/180)$ ” 返回 1.732050808； $\text{TAN}(1)$ 返回 1.557407725。

[59]. TANH

用途：返回任意实数的双曲正切值。

语法： $\text{TANH}(\text{number})$

参数：Number 为任意实数。

实例：如果 $A1=60$ ，则公式 “ $=\text{TANH}(A1)$ ” 返回 1， $=\text{TANH}(0.5)$ 返回 0.462117。

[60]. TRUNC

用途：将数字的小数部分截去，返回整数。

语法： $\text{TRUNC}(\text{number}, \text{num_digits})$

参数：Number 是需要截去小数部分的数字，Num_digits 则指定保留小数的精度(几位小数)。注意：TRUNC 函数可以按需要截取数字的小数部分，而 INT 函数则将数字向下舍入到最接近的整数。INT 和 TRUNC 函数在处理负数时有所不同：TRUNC(-4.3) 返回 -4，而 INT(-4.3) 返回 -5。

实例：如果 $A1=78.652$ ，则公式 “ $=\text{TRUNC}(A1, 1)$ ” 返回 78.6， $=\text{TRUNC}(A1, 2)$ 返回 78.65， $=\text{TRUNC}(-8.963, 2)$ 返回 -8.96。

10. 统计函数

[1]. AVEDEV

用途：返回一组数据与其平均值的绝对偏差的平均值，该函数可以评测数据(例如学生的某科考试成绩)的离散度。

语法： $\text{AVEDEV}(\text{number1}, \text{number2}, \dots)$

参数：Number1、number2、... 是用来计算绝对偏差平均值的一组参数，其个数可以在 1~30 个之间。

实例：如果 $A1=79$ 、 $A2=62$ 、 $A3=45$ 、 $A4=90$ 、 $A5=25$ ，则公式 “ $=\text{AVEDEV}(A1:A5)$ ” 返回 20.16。

[2]. AVERAGE

用途：计算所有参数的算术平均值。

语法： $\text{AVERAGE}(\text{number1}, \text{number2}, \dots)$

参数：Number1、number2、... 是要计算平均值的 1~30 个参数。

实例：如果 A1:A5 区域命名为分数，其中的数值分别为 100、70、92、47 和 82，则公式 “ $=\text{AVERAGE}(\text{分数})$ ” 返回 78.2。

[3]. AVERAGEA

用途：计算参数清单中数值的平均值。它与 AVERAGE 函数的区别在于不仅数字，而

且文本和逻辑值(如 TRUE 和 FALSE)也参与计算。

语法: AVERAGEA(value1, value2, ...)

参数: Value1、value2、...为需要计算平均值的 1 至 30 个单元格、单元格区域或数值。

实例: 如果 A1=76、A2=85、A3=TRUE, 则公式 “=AVERAGEA(A1:A3)” 返回 54(即 $76+85+1/3=54$)。

[4]. BETADIST

用途: 返回 Beta 分布累积函数的函数值。Beta 分布累积函数通常用于研究样本集合中某些事物的发生和变化情况。例如, 人们一天中看电视的时间比率。

语法: BETADIST(x, alpha, beta, A, B)

参数: X 用来进行函数计算的值, 须居于可选性上下界(A 和 B)之间。Alpha 分布的参数。Beta 分布的参数。A 是数值 x 所属区间的可选下界, B 是数值 x 所属区间的可选上界。

实例: 公式 “=BETADIST(2, 8, 10, 1, 3)” 返回 0.685470581。

[5]. BETAINV

用途: 返回 beta 分布累积函数的逆函数值。即, 如果 $\text{probability}=\text{BETADIST}(x, \dots)$, 则 $\text{BETAINV}(\text{probability}, \dots)=x$ 。beta 分布累积函数可用于项目设计, 在给出期望的完成时间和变化参数后, 模拟可能的完成时间。

语法: BETAINV(probability, alpha, beta, A, B)

参数: Probability 为 Beta 分布的概率值, Alpha 分布的参数, Beta 分布的参数, A 数值 x 所属区间的可选下界, B 数值 x 所属区间的可选上界。

实例: 公式 “=BETAINV(0.685470581, 8, 10, 1, 3)” 返回 2。

[6]. BINOMDIST

用途: 返回一元二项式分布的概率值。BINOMDIST 函数适用于固定次数的独立实验, 实验的结果只包含成功或失败二种情况, 且成功的概率在实验期间固定不变。例如, 它可以计算掷 10 次硬币时正面朝上 6 次的概率。

语法: BINOMDIST(number_s, trials, probability_s, cumulative)

参数: Number_s 为实验成功的次数, Trials 为独立实验的次数, Probability_s 为一次实验中成功的概率, Cumulative 是一个逻辑值, 用于确定函数的形式。如果 cumulative 为 TRUE, 则 BINOMDIST 函数返回累积分布函数, 即至多 number_s 次成功的概率; 如果为 FALSE, 返回概率密度函数, 即 number_s 次成功的概率。

实例: 抛硬币的结果不是正面就是反面, 第一次抛硬币为正面的概率是 0.5。则掷硬币 10 次中 6 次的计算公式为 “=BINOMDIST(6, 10, 0.5, FALSE)”, 计算的结果等于 0.205078

[7]. CHIDIST

用途: 返回 χ^2 分布的单尾概率。 χ^2 分布与 χ^2 检验相关。使用 χ^2 检验可以比较观察值和期望值。例如, 某项遗传学实验假设下一代植物将呈现出某一组颜色。使用此函数比较观测结果和期望值, 可以确定初始假设是否有效。

语法: CHIDIST(x, degrees_freedom)

参数: X 是用来计算 c2 分布单尾概率的数值, Degrees_freedom 是自由度。

实例: 公式 “=CHIDIST(1, 2)” 的计算结果等于 0.606530663。

[8]. CHIINV

用途: 返回 c2 分布单尾概率的逆函数。如果 probability=CHIDIST(x, ?), 则 CHIINV(probability, ?)=x。使用此函数比较观测结果和期望值, 可以确定初始假设是否有效。

语法: CHIINV(probability, degrees_freedom)

参数: Probability 为 c2 分布的单尾概率, Degrees_freedom 为自由度

实例: 公式 “=CHIINV(0.5, 2)” 返回 1.386293564。

[9]. CHITEST

用途: 返回相关性检验值, 即返回 c2 分布的统计值和相应的自由度, 可使用 c2 检验确定假设值是否被实验所证实。

语法: CHITEST(actual_range, expected_range)

参数: Actual_range 是包含观察值的数据区域, Expected_range 是包含行列汇总的乘积与总计值之比的数据区域。

实例: 如果 A1=1、A2=2、A3=3、B1=4、B2=5、B3=6, 则公式 “=CHITEST(A1:A3, B1:B3)” 返回 0.062349477。

[10]. CONFIDENCE

用途: 返回总体平均值的置信区间, 它是样本平均值任意一侧的区域。例如, 某班学生参加考试, 依照给定的置信度, 可以确定该次考试的最低和最高分数。

语法: CONFIDENCE(alpha, standard_dev, size)

参数: Alpha 是用于计算置信度(它等于 100*(1-alpha)%, 如果 alpha 为 0.05, 则置信度为 95%)的显著水平参数, Standard_dev 是数据区域的总体标准偏差, Size 为样本容量。

实例: 假设样本取自 46 名学生的考试成绩, 他们的平均分为 60, 总体标准偏差为 5 分, 则平均分在下列区域内的置信度为 95%。公式 “=CONFIDENCE(0.05, 5, 46)” 返回 1.44, 即考试成绩为 60±1.44 分。

[11]. CORREL

用途: 返回单元格区域 array1 和 array2 之间的相关系数。它可以确定两个不同事物之间的关系, 例如检测学生的物理与数学学习成绩之间是否关联。

语法: CORREL(array1, array2)

参数: Array1 第一组数值单元格区域。Array2 第二组数值单元格区域。

实例: 如果 A1=90、A2=86、A3=65、A4=54、A5=36、B1=89、B2=83、B3=60、B4=50、B5=32, 则公式 “=CORREL(A1:A5, B1:B5)” 返回 0.998876229, 可以看出 A、B 两列数据具有很高的相关性。

[12]. COUNT

用途: 返回数字参数的个数。它可以统计数组或单元格区域中含有数字的单元格个数。

语法: COUNT(value1, value2, ...)

参数: Value1, value2, ... 是包含或引用各种类型数据的参数(1~30 个), 其中只有数字类型的数据才能被统计。

实例: 如果 A1=90、A2=人数、A3=" "、A4=54、A5=36, 则公式 “=COUNT(A1:A5)” 返回 3。

[13]. COUNTA

用途: 返回参数组中非空值的数目。利用函数 COUNTA 可以计算数组或单元格区域中数据项的个数。

语法: COUNTA(value1, value2, ...) 说明: Value1, value2, ... 所要计数的值, 参数个数为 1~30 个。在这种情况下, 参数可以是任何类型, 它们包括空格但不包括空白单元格。如果参数是数组或单元格引用, 则数组或引用中的空白单元格将被忽略。如果不需要统计逻辑值、文字或错误值, 则应该使用 COUNT 函数。

实例: 如果 A1=6.28、A2=3.74, 其余单元格为空, 则公式 “=COUNTA(A1:A7)” 的计算结果等于 2。

[14]. COUNTBLANK

用途: 计算某个单元格区域中空白单元格的数目。

语法: COUNTBLANK(range)

参数: Range 为需要计算其中空白单元格数目的区域。

实例: 如果 A1=88、A2=55、A3=""、A4=72、A5="", 则公式 “=COUNTBLANK(A1:A5)” 返回 2。

[15]. COUNTIF

用途: 计算区域中满足给定条件的单元格的个数。

语法: COUNTIF(range, criteria)

参数: Range 为需要计算其中满足条件的单元格数目的单元格区域。Criteria 为确定哪些单元格将被计算在内的条件, 其形式可以为数字、表达式或文本。

[16]. COVAR

用途: 返回协方差, 即每对数据点的偏差乘积的平均数。利用协方差可以研究两个数据集之间的关系。

语法: COVAR(array1, array2)

参数: Array1 是第一个所含数据为整数的单元格区域, Array2 是第二个所含数据为整数的单元格区域。

实例: 如果 A1=3、A2=2、A3=1、B1=3600、B2=1500、B3=800, 则公式 “=COVAR(A1:A3, B1:B3)” 返回 933.3333333。

[17]. CRITBINOM

用途：返回使累积二项式分布大于等于临界值的最小值，其结果可以用于质量检验。例如决定最多允许出现多少个有缺陷的部件，才可以保证当整个产品在离开装配线时检验合格。

语法：CRITBINOM(trials, probability_s, alpha)

参数：Trials 是伯努利实验的次数，Probability_s 是一次试验中成功的概率，Alpha 是临界值。

实例：公式 “=CRITBINOM(10, 0.9, 0.75)” 返回 10。

[18]. DEVSQ

用途：返回数据点与各自样本平均值的偏差的平方和。

语法：DEVSQ(number1, number2, ...)

参数：Number1、number2、...是用于计算偏差平方和的 1 到 30 个参数。它们可以用逗号分隔的数值，也可以是数组引用。

实例：如果 A1=90、A2=86、A3=65、A4=54、A5=36，则公式 “=DEVSQ(A1: A5)” 返回 2020.8。

[19]. EXPONDIST

用途：返回指数分布。该函数可以建立事件之间的时间间隔模型，如估计银行的自动取款机支付一次现金所花费的时间，从而确定此过程最长持续一分钟的发生概率。

语法：EXPONDIST(x, lambda, cumulative)

参数：X 函数的数值，Lambda 参数值，Cumulative 为确定指数函数形式的逻辑值。如果 cumulative 为 TRUE，EXPONDIST 返回累积分布函数；如果 cumulative 为 FALSE，则返回概率密度函数。

实例：公式 “=EXPONDIST(0.2, 10, TRUE)” 返回 0.864665，=EXPONDIST(0.2, 10, FALSE)返回 1.353353。

[20]. FDIST

用途：返回 F 概率分布，它可以确定两个数据系列是否存在变化程度上的不同。例如，通过分析某一班级男、女生的考试分数，确定女生分数的变化程度是否与男生不同。

语法：FDIST(x, degrees_freedom1, degrees_freedom2)

参数：X 是用来计算概率分布的区间点，Degrees_freedom1 是分子自由度，Degrees_freedom2 是分母自由度。

实例：公式 “=FDIST(1, 90, 89)” 返回 0.500157305。

[21]. FINV

用途：返回 F 概率分布的逆函数值，即 F 分布的临界值。如果 $p = \text{FDIST}(x, .)$ ，则 $\text{FINV}(p, .) = x$ 。

语法：FINV(probability, degrees_freedom1, degrees_freedom2)

参数：Probability 是累积 F 分布的概率值，Degrees_freedom1 是分子自由度，

Degrees_freedom2 是分母自由度。

实例：公式 “=FINV(0.1, 86, 74)” 返回 1.337888023。

[22]. FISHER

用途：返回点 x 的 Fisher 变换。该变换生成一个近似正态分布而非偏斜的函数，使用此函数可以完成相关系数的假设性检验。

语法：FISHER(x)

参数：X 为一个数字，在该点进行变换。

实例：公式 “=FISHER(0.55)” 返回 0.618381314。

[23]. FISHERINV

用途：返回 Fisher 变换的逆函数值，如果 $y = \text{FISHER}(x)$ ，则 $\text{FISHERINV}(y) = x$ 。上述变换可以分析数据区域或数组之间的相关性。

语法：FISHERINV(y)

参数：Y 为一个数值，在该点进行反变换。

实例：公式 “=FISHERINV(0.765)” 返回 0.644012628。

[24]. FORECAST

用途：根据一条线性回归拟合线返回一个预测值。使用此函数可以对未来销售额、库存需求或消费趋势进行预测。

语法：FORECAST(x, known_y's, known_x's)

参数：X 为需要进行预测的数据点的 X 坐标(自变量值)。Known_y's 是从满足线性拟合直线 $y = kx + b$ 的点集中选出的一组已知的 y 值，Known_x's 是从满足线性拟合直线 $y = kx + b$ 的点集中选出的一组已知的 x 值。

实例：公式 “=FORECAST(16, {7, 8, 9, 11, 15}, {21, 26, 32, 36, 42})” 返回 4.378318584。

[25]. FREQUENCY

用途：以一系列垂直数组返回某个区域中数据的频率分布。它可以计算出在给定的值域和接收区间内，每个区间包含的数据个数。

语法：FREQUENCY(data_array, bins_array)

参数：Data_array 是用来计算频率一个数组，或对数组单元区域的引用。Bins_array 是数据接收区间，为一数组或对数组区域的引用，设定对 data_array 进行频率计算的分段点。

[26]. FTEST

用途：返回 F 检验的结果。它返回的是当数组 1 和数组 2 的方差无明显差异时的单尾概率，可以判断两个样本的方差是否不同。例如，给出两个班级同一学科考试成绩，从而检验是否存在差别。

语法：FTEST(array1, array2)

参数：Array1 是第一个数组或数据区域，Array2 是第二个数组或数据区域。

实例：如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，B1=59、B2=70、B3=80、B4=90、B5=89、B6=84、B7=92，则公式 “=FTEST(A1:A7, B1:B7)” 返回 0.519298931。

[27]. GAMMADIST

用途：返回伽玛分布。可用它研究具有偏态分布的变量，通常用于排队分析。

语法：GAMMADIST(x, alpha, beta, cumulative)

参数：X 为用来计算伽玛分布的数值，Alpha 是 γ 分布参数，Beta 分布的一个参数。如果 beta=1，GAMMADIST 函数返回标准伽玛分布。Cumulative 为一逻辑值，决定函数的形式。如果 cumulative 为 TRUE，GAMMADIST 函数返回累积分布函数；如果为 FALSE，则返回概率密度函数。

实例：公式 “=GAMMADIST(10, 9, 2, FALSE)” 的计算结果等于 0.032639，=GAMMADIST(10, 9, 2, TRUE) 返回 0.068094。

[28]. GAMMAINV

用途：返回具有给定概率的伽玛分布的区间点，用来研究出现分布偏斜的变量。如果 $P = \text{GAMMADIST}(x, \dots)$ ，则 $\text{GAMMAINV}(p, \dots) = x$ 。

语法：GAMMAINV(probability, alpha, beta)

参数：Probability 为伽玛分布的概率值，Alpha 分布参数，Beta 分布参数。如果 beta=1，函数 GAMMAINV 返回标准伽玛分布。

实例：公式 “=GAMMAINV(0.05, 8, 2)” 返回 7.96164386。

[29]. GAMMALN

用途：返回伽玛函数的自然对数 $\Gamma(x)$ 。

语法：GAMMALN(x)

参数：X 为需要计算 GAMMALN 函数的数值。

实例：公式 “=GAMMALN(6)” 返回 4.787491743。

[30]. GEOMEAN

用途：返回正数数组或数据区域的几何平均值。可用于计算可变复利的平均增长率。

语法：GEOMEAN(number1, number2, ...)

参数：Number1, number2, ... 为需要计算其平均值的 1 到 30 个参数，除了使用逗号分隔数值的形式外，还可使用数组或对数组的引用。

实例：公式 “=GEOMEAN(1.2, 1.5, 1.8, 2.3, 2.6, 2.8, 3)” 的计算结果是 2.069818248。

[31]. GROWTH

用途：给定的数据预测指数增长值。根据已知的 x 值和 y 值，函数 GROWTH 返回一组新的 x 值对应的 y 值。通常使用 GROWTH 函数拟合满足给定 x 值和 y 值的指数曲线。

语法：GROWTH(known_y's, known_x's, new_x's, const)

参数：Known_y's 是满足指数回归拟合曲线 $y = b * m^x$ 的一组已知的 y 值；Known_x's 是满足指数回归拟合曲线 $y = b * m^x$ 的一组已知的 x 值的集合(可选参数)；New_x's 是

组新的 x 值，可通过 GROWTH 函数返回各自对应的 y 值；Const 为一逻辑值，指明是否将系数 b 强制设为 1，如果 const 为 TRUE 或省略，b 将参与正常计算。如果 const 为 FALSE，b 将被设为 1，m 值将被调整使得 $y=m^x$ 。

[32]. HARMEAN

用途：返回数据集合的调和平均值。调和平均值与倒数的算术平均值互为倒数。调和平均值总小于几何平均值，而几何平均值总小于算术平均值。

语法：HARMEAN(number1, number2, ...)

参数：Number1, number2, ...是需要计算其平均值的 1 到 30 个参数。可以使用逗号分隔参数的形式，还可以使用数组或数组的引用。

实例：公式 “=HARMEAN(66, 88, 92)” 返回 80.24669604。

[33]. HYPGEOMDIST

用途：返回超几何分布。给定样本容量、样本总体容量和样本总体中成功的次数，HYPGEOMDIST 函数返回样本取得给定成功次数的概率。

语 法： HYPGEOMDIST(sample_s, number_sample, population_s, number_population)

参数：Sample_s 为样本中成功的次数，Number_sample 为样本容量。Population_s 为样本总体中成功的次数，Number_population 为样本总体的容量。

实例：如果某个班级有 42 名学生。其中 22 名是男生，20 名是女生。如果随机选出 6 人，则其中恰好有三名女生的概率公式是：“=HYPGEOMDIST(3, 6, 20, 42)”，返回的结果为 0.334668627。

[34]. INTERCEPT

用途：利用已知的 x 值与 y 值计算直线与 y 轴的截距。当已知自变量为零时，利用截距可以求得因变量的值。

语法：INTERCEPT(known_y's, known_x's)

参数：Known_y's 是一组因变量数据或数据组，Known_x's 是一组自变量数据或数据组。

实例：如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，B1=59、B2=70、B3=80、B4=90、B5=89、B6=84、B7=92，则公式 “=INTERCEPT(A1:A7, B1:B7)” 返回 87.61058785。

[35]. KURT

用途：返回数据集的峰值。它反映与正态分布相比时某一分布的尖锐程度或平坦程度，正峰值表示相对尖锐的分布，负峰值表示相对平坦的分布。

语法：KURT(number1, number2, ...)

参数：Number1, number2, ...为需要计算其峰值的 1 到 30 个参数。它们可以使用逗号分隔参数的形式，也可以使用单一数组，即对数组单元格的引用。

实例：如果某次学生考试的成绩为 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，则公式 “=KURT(A1:A7)” 返回 -1.199009798，说明这次的成绩相对正态分布是一比较平坦的分布。

[36]. LARGE

用途：返回某一数据集中的某个最大值。可以使用 LARGE 函数查询考试分数集中第第三等的得分。

语法：LARGE(array, k)

参数：Array 为需要从中查询第 k 个最大值的数组或数据区域，K 为返回值在数组或数据单元格区域里的位置(即名次)。

实例：如果 B1=59、B2=70、B3=80、B4=90、B5=89、B6=84、B7=92,, 则公式 “=LARGE(B1, B7, 2)” 返回 90。

[37]. LINEST

用途：使用最小二乘法对已知数据进行最佳直线拟合，并返回描述此直线的数组。

语法：LINEST(known_y's, known_x's, const, stats)

参数：Known_y's 是表达式 $y=mx+b$ 中已知的 y 值集合，Known_x's 是关系表达式 $y=mx+b$ 中已知的可选 x 值集合，Const 为一逻辑值，指明是否强制使常数 b 为 0，如果 const 为 TRUE 或省略，b 将参与正常计算。如果 const 为 FALSE，b 将被设为 0，并同时调整 m 值使得 $y=mx$ 。Stats 为一逻辑值，指明是否返回附加回归统计值。如果 stats 为 TRUE，函数 LINEST 返回附加回归统计值。如果 stats 为 FALSE 或省略，函数 LINEST 只返回系数 m 和常数项 b。

实例：如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，B1=59、B2=70、B3=80、B4=90、B5=89、B6=84、B7=92，则数组公式 “{=LINEST(A1:A7, B1:B7)}” 返回 -0.174244885、-0.174244885、-0.174244885、-0.174244885、-0.174244885、-0.174244885、-0.174244885。

[38]. LOGEST

用途：在回归分析中，计算最符合观测数据组的指数回归拟合曲线，并返回描述该曲线的数组。

语法：LOGEST(known_y's, known_x's, const, stats)

参数：Known_y's 是一组符合 $y=b*m^x$ 函数关系的 y 值的集合，Known_x's 是一组符合 $y=b*m^x$ 运算关系的可选 x 值集合，Const 是指定是否要设定常数 b 为 1 的逻辑值，如果 const 设定为 TRUE 或省略，则常数项 b 将通过计算求得。

实例：如果某公司的新产品销售额呈指数增长，依次为 A1=33100、A2=47300、A3=69000、A4=102000、A5=150000 和 A6=220000，同时 B1=11、B2=12、B3=13、B4=14、B5=15、B6=16。则使用数组公式 “{=LOGEST(A1:A6, B1:B6, TRUE, TRUE)}”，在 C1:D5 单元格内得到的计算结果是：1.463275628、495.3047702、0.002633403、0.035834282、0.99980862、0.011016315、20896.8011、4、2.53601883 和 0.000485437。

[39]. LOGINV

用途：返回 x 的对数正态分布累积函数的逆函数，此处的 $\ln(x)$ 是含有 mean(平均数)与 standard-dev(标准差)参数的正态分布。如果 $p=\text{LOGNORMDIST}(x, \dots)$ ，那么 $\text{LOGINV}(p, \dots)=x$ 。

语法：LOGINV(probability, mean, standard_dev)

参数：Probability 是与对数正态分布相关的概率，Mean 为 $\ln(x)$ 的平均数，Standard_dev 为 $\ln(x)$ 的标准偏差。

实例：公式 “=LOGINV(0.036, 2.5, 1.5)” 返回 0.819815949。

[40]. LOGNORMDIST

用途：返回 x 的对数正态分布的累积函数，其中 $\ln(x)$ 是服从参数为 mean 和 standard_dev 的正态分布。使用此函数可以分析经过对数变换的数据。

语法：LOGNORMDIST(x , mean, standard_dev)

参数：X 是用来计算函数的数值，Mean 是 $\ln(x)$ 的平均值，Standard_dev 是 $\ln(x)$ 的标准偏差。

实例：公式 “=LOGNORMDIST(2, 5.5, 1.6)” 返回 0.001331107。

[41]. MAX

用途：返回数据集中的最大数值。

语法：MAX(number1, number2, ...)

参数：Number1, number2, ...是需要找出最大数值的 1 至 30 个数值。

实例：如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，则公式 “=MAX(A1:A7)” 返回 96。

[42]. MAXA

用途：返回数据集中的最大数值。它与 MAX 的区别在于文本值和逻辑值(如 TRUE 和 FALSE)作为数字参与计算。

语法：MAXA(value1, value2, ...)

参数：Value1, Value2, ...为需要从中查找最大数值的 1 到 30 个参数。例：如果 A1:A5 包含 0、0.2、0.5、0.4 和 TRUE，则：MAXA(A1:A5)返回 1。

[43]. MEDIAN

用途：返回给定数值集合的中位数(它是在一组数据中居于中间的数。换句话说，在这组数据中，有一半的数据比它大，有一半的数据比它小)

语法：MEDIAN(number1, number2, ...)

参数：Number1, number2, ...是需要找出中位数的 1 到 30 个数字参数。

实例：MEDIAN(11, 12, 13, 14, 15)返回 13；MEDIAN(1, 2, 3, 4, 5, 6)返回 3.5，即 3 与 4 的平均值。

[44]. MIN

用途：返回给定参数表中的最小值。

语法：MIN(number1, number2, ...)

参数：Number1, number2, ...是要从中找出最小值的 1 到 30 个数字参数。

实例：如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=96，则公式 “=MIN(A1:A7)” 返回 49；而=MIN(A1:A5, 0, -8)返回-8。

[45]. MINA

用途: 返回参数清单中的最小数值。它与 MIN 函数的区别在于文本值和逻辑值(如 TRUE 和 FALSE)也作为数字参与计算。

语法: MINA(value1, value2, ...)

参数: Value1, value2, ...为需要从中查找最小数值的 1 到 30 个参数。

实例: 如果 A1=71、A2=83、A3=76、A4=49、A5=92、A6=88、A7=FALSE, 则公式 “=MINA(A1:A7)” 返回 0。

[46]. MODE

用途: 返回在某一数组或数据区域中的众数。

语法: MODE(number1, number2, ...)

参数: Number1, number2, ...是用于众数计算的 1 到 30 个参数。

实例: 如果 A1=71、A2=83、A3=71、A4=49、A5=92、A6=88, 则公式 “=MODE(A1:A6)” 返回 71。

[47]. NEGBINOMDIST

用途: 返回负二项式分布。当成功概率为常数 probability_s 时, 函数 NEGBINOMDIST 返回在到达 number_s 次成功之前, 出现 number_f 次失败的概率。此函数与二项式分布相似, 只是它的成功次数固定, 试验总数为变量。与二项分布类似的是, 试验次数被假设为自变量。

语法: NEGBINOMDIST(number_f, number_s, probability_s), Number_f 是失败次数, Number_s 为成功的临界次数, Probability_s 是成功的概率。

实例: 如果要找 10 个反应敏捷的人, 且已知具有这种特征的候选人的概率为 0.3。那么, 找到 10 个合格候选人之前, 需要对不合格候选人进行面试的概率公式为 “=NEGBINOMDIST(40, 10, 0.3)”, 计算结果是 0.007723798。

[48]. NORMDIST

用途: 返回给定平均值和标准偏差的正态分布的累积函数。

语法: NORMDIST(x, mean, standard_dev, cumulative)

参数: X 为用于计算正态分布函数的区间点, Mean 是分布的算术平均值, Standard_dev 是分布的标准方差; Cumulative 为一逻辑值, 指明函数的形式。如果 cumulative 为 TRUE, 则 NORMDIST 函数返回累积分布函数; 如果为 FALSE, 则返回概率密度函数。

实例: 公式 “=NORMDIST(46, 35, 2.5, TRUE)” 返回 0.999994583。

[49]. NORMSINV

用途: 返回标准正态分布累积函数的逆函数。该分布的平均值为 0, 标准偏差为 1。

语法: NORMSINV(probability)

参数: Probability 是正态分布的概率值。

实例: 公式 “=NORMSINV(0.8)” 返回 0.841621386。

[50]. NORMSDIST

用途：返回标准正态分布的累积函数，该分布的平均值为 0，标准偏差为 1。

语法：NORMSDIST(z)

参数：Z 为需要计算其分布的数值。

实例：公式 “=NORMSDIST(1.5)” 的计算结果为 0.933192771。

[51]. NORMSINV

用途：返回标准正态分布累积函数的逆函数。该分布的平均值为 0，标准偏差为 1。

语法：NORMSINV(probability)

参数：Probability 是正态分布的概率值。

实例：公式 “=NORMSINV(0.933192771)” 返回 1.499997779(即 1.5)。

[52]. PEARSON

用途：返回 Pearson(皮尔生)乘积矩相关系数 r，它是一个范围在-1.0 到 1.0 之间(包括-1.0 和 1.0 在内)的无量纲指数，反映了两个数据集合之间的线性相关程度。

语法：PEARSON(array1, array2)

参数：Array1 为自变量集合，Array2 为因变量集合。

实例：如果 A1=71、A2=83、A3=71、A4=49、A5=92、A6=88，B1=69、B2=80、B3=76、B4=40、B5=90、B6=81，则公式 “=PEARSON(A1:A6,B1:B6)” 返回 0.96229628。

[53]. PERCENTILE

用途：返回数值区域的 K 百分比数值点。例如确定考试排名在 80 个百分点以上的分数。

语法：PERCENTILE(array, k)

参数：Array 为定义相对位置的数值数组或数值区域，k 为数组中需要得到其排位的值。

实例：如果某次考试成绩为 A1=71、A2=83、A3=71、A4=49、A5=92、A6=88，公式 “=PERCENTILE(A1:A6, 0.8)” 返回 88，即考试排名要想在 80 个百分点以上，则分数至少应当为 88 分。

[54]. PERCENTRANK

用途：返回某个数值在一个数据集中的百分比排位，可用于查看数据在数据集中所处的位置。例如计算某个分数在所有考试成绩中所处的位置。

语法：PERCENTRANK(array, x, significance)

参数：Array 为彼此间相对位置确定的数据集合，X 为其中需要得到排位的值，Significance 为可选项，表示返回的百分数值的有效位数。如果省略，函数 PERCENTRANK 保留 3 位小数。

实例：如果某次考试成绩为 A1=71、A2=83、A3=71、A4=49、A5=92、A6=88，则公式 “=PERCENTRANK(A1:A6, 71)” 的计算结果为 0.2，即 71 分在 6 个分数中排 20%。

[55]. PERMUT

用途：返回从给定数目的元素集合中选取的若干元素的排列数。

语法：PERMUT(number, number_chosen)

参数：Number 为元素总数，Number_chosen 是每个排列中的元素数目。

实例：如果某种彩票的号码有 9 个数，每个数的范围是从 0 到 9(包括 0 和 9)。则所有可能的排列数量用公式 “=PERMUT(10, 9)” 计算，其结果为 3628800。

[56]. POISSON

用途：返回泊松分布。泊松分布通常用于预测一段时间内事件发生的次数，比如一分钟内通过收费站的轿车的数量。

语法：POISSON(x, mean, cumulative)

参数：X 是某一事件出现的次数，Mean 是期望值，Cumulative 为确定返回的概率分布形式的逻辑值。

实例：公式 “=POISSON(5, 10, TRUE)” 返回 0.067085963, =POISSON(3, 12, FALSE) 返回 0.001769533。

[57]. PROB

用途：返回一概率事件组中落在指定区域内的事件所对应的概率之和。

语法：PROB(x_range, prob_range, lower_limit, upper_limit)

参数：X_range 是具有各自相应概率值的 x 数值区域，Prob_range 是与 x_range 中的数值相对应的一组概率值，Lower_limit 是用于概率求和计算的数值下界，Upper_limit 是用于概率求和计算的数值可选上界。

实例：公式 “=PROB({0, 1, 2, 3}, {0.2, 0.3, 0.1, 0.4}, 2)” 返回 0.1, =PROB({0, 1, 2, 3}, {0.2, 0.3, 0.1, 0.4}, 1, 3) 返回 0.8。

[58]. QUARTILE

用途：返回一组数据的四分位点。四分位数通常用于在考试成绩之类的数据集中对总体进行分组，如求出一组分数中前 25% 的分数。

语法：QUARTILE(array, quart)

参数：Array 为需要求得四分位数值的数据或数字引用区域，Quart 决定返回哪一个四分位值。如果 quart 取 0、1、2、3 或 4，则函数 QUARTILE 返回最小值、第一个四分位数(第 25 个百分排位)、中分位数(第 50 个百分排位)、第三个四分位数(第 75 个百分排位)和最大数值。

实例：如果 A1=78、A2=45、A3=90、A4=12、A5=85，则公式 “=QUARTILE(A1:A5, 3)” 返回 85。

[59]. RANK

用途：返回一个数值在一组数值中的排位(如果数据清单已经排过序了，则数值的排位就是它当前的位置)。

语法：RANK(number, ref, order)

参数: Number 是需要计算其排位的一个数字; Ref 是包含一组数字的数组或引用(其中的非数值型参数将被忽略); Order 为一数字, 指明排位的方式。如果 order 为 0 或省略, 则按降序排列的数据清单进行排位。如果 order 不为零, ref 当作按升序排列的数据清单进行排位。注意: 函数 RANK 对重复数值的排位相同。但重复数的存在将影响后续数值的排位。如在一列整数中, 若整数 60 出现两次, 其排位为 5, 则 61 的排位为 7(没有排位为 6 的数值)。

实例: 如果 A1=78、A2=45、A3=90、A4=12、A5=85, 则公式 “=RANK(A1, \$A\$1:\$A\$5)” 返回 5、8、2、10、4。

[60]. RSQ

用途: 返回给定数据点的 Pearson 乘积矩相关系数的平方。

语法: RSQ(known_y's, known_x's)

参数: Known_y's 为一个数组或数据区域, Known_x's 也是一个数组或数据区域。

实例: 公式 “=RSQ({22, 23, 29, 19, 38, 27, 25}, {16, 15, 19, 17, 15, 14, 34})” 返回 0.013009334。

[61]. SKEW

用途: 返回一个分布的不对称度。它反映以平均值为中心的分布的不对称程度, 正不对称度表示不对称边的分布更趋向正值。负不对称度表示不对称边的分布更趋向负值。

语法: SKEW(number1, number2, ...)

参数: Number1, number2...是需要计算不对称度的 1 到 30 个参数。包括逗号分隔的数值、单一数组和名称等。

实例: 公式 “=SKEW({22, 23, 29, 19, 38, 27, 25}, {16, 15, 19, 17, 15, 14, 34})” 返回 0.854631382。

[62]. SLOPE

用途: 返回经过给定数据点的线性回归拟合线方程的斜率(它是直线上任意两点的垂直距离与水平距离的比值, 也就是回归直线的变化率)。

语法: SLOPE(known_y's, known_x's)

参数: Known_y's 为数字型因变量数组或单元格区域, Known_x's 为自变量数据点集合。

实例: 公式 “=SLOPE({22, 23, 29, 19, 38, 27, 25}, {16, 15, 19, 17, 15, 14, 34})” 返回-0.100680934。

[63]. SMALL

用途: 返回数据集中第 k 个最小值, 从而得到数据集中特定位置上的数值。

语法: SMALL(array, k)

参数: Array 是需要找到第 k 个最小值的数组或数字型数据区域, K 为返回的数据在数组或数据区域里的位置(从小到大)。

实例: 如果 A1=78、A2=45、A3=90、A4=12、A5=85, 则公式 “=SMALL(A1:A5, 3)” 返回 78。

[64]. STANDARDIZE

用途：返回以 mean 为平均值，以 standard-dev 为标准偏差的分布的正态化数值。

语法：STANDARDIZE(x, mean, standard_dev)

参数：X 为需要进行正态化的数值，Mean 分布的算术平均值，Standard_dev 为分布的标准偏差。

实例：公式 “=STANDARDIZE(62, 60, 10)” 返回 0.2。

[65]. STDEV

用途：估算样本的标准偏差。它反映了数据相对于平均值(mean)的离散程度。

语法：STDEV(number1, number2, ...)

参数：Number1, number2, ...为对应于总体样本的 1 到 30 个参数。可以使用逗号分隔的参数形式，也可使用数组，即对数组单元格的引用。注意：STDEV 函数假设其参数是总体中的样本。如果数据是全部样本总体，则应该使用 STDEVP 函数计算标准偏差。同时，函数忽略参数中的逻辑值(TRUE 或 FALSE)和文本。如果不能忽略逻辑值和文本，应使用 STDEVA 函数。

实例：假设某次考试的成绩样本为 A1=78、A2=45、A3=90、A4=12、A5=85，则估算所有成绩标准偏差的公式为 “=STDEV(A1:A5)”，其结果等于 33.00757489。

[66]. STDEVA

用途：计算基于给定样本的标准偏差。它与 STDEV 函数的区别是文本值和逻辑值(TRUE 或 FALSE)也将参与计算。

语法：STDEVA(value1, value2, ...)

参数：Value1, value2, ...是作为总体样本的 1 到 30 个参数。可以使用逗号分隔参数的形式，也可以使用单一数组，即对数组单元格的引用。

实例：假设某次考试的部分成绩为 A1=78、A2=45、A3=90、A4=12、A5=85，则估算所有成绩标准偏差的公式为 “=STDEVA(A1:A5)”，其结果等于 33.00757489。

[67]. STDEVP

用途：返回整个样本总体的标准偏差。它反映了样本总体相对于平均值(mean)的离散程度。

语法：STDEVP(number1, number2, ...)

参数：Number1, number2, ...为对应于样本总体的 1 到 30 个参数。可以使用逗号分隔参数的形式，也可以使用单一数组，即对数组单元格的引用。注意：STDEVP 函数在计算过程中忽略逻辑值(TRUE 或 FALSE)和文本。如果逻辑值和文本不能忽略，应当使用 STDEVP 函数。同时 STDEVP 函数假设其参数为整个样本总体。如果数据代表样本总体中的样本，应使用函数 STDEV 来计算标准偏差。当样本数较多时，STDEV 和 STDEVP 函数的计算结果相差很小。

实例：如果某次考试只有 5 名学生参加，成绩为 A1=78、A2=45、A3=90、A4=12、A5=85，则计算的所有成绩的标准偏差公式为 “=STDEVP(A1:A5)”，返回的结果等于 29.52287249。

[68]. STDEVPA

用途：计算样本总体的标准偏差。它与 STDEVP 函数的区别是文本值和逻辑值(TRUE 或 FALSE)参与计算。

语法：STDEVPA(value1, value2, ...)

参数：Value1, value2, ...作为样本总体的 1 到 30 个参数。可以使用逗号分隔参数的形式，也可以使用单一数组(即对数组单元格的引用)。注意：STDEVPA 函数假设参数为样本总体。如果数据代表的是总体的部分样本，则必须使用 STDEVA 函数来估算标准偏差。

实例：如果某次考试只有 5 名学生参加，成绩为 A1=78、A2=45、A3=90、A4=12、A5=85，则计算的所有成绩的标准偏差公式为“=STDEVP(A1:A5)”，返回的结果等于 29.52287249。

[69]. STEYX

用途：返回通过线性回归法计算 y 预测值时所产生的标准误差。标准误差用来度量根据单个 x 变量计算出的 y 预测值的误差量。

语法：STEYX(known_y's, known_x's)

参数：Known_y's 为因变量数据点数组或区域，Known_x's 为自变量数据点数组或区域。

实例：公式“=STEYX({22, 13, 29, 19, 18, 17, 15}, {16, 25, 11, 17, 25, 14, 17})”返回 4.251584755。

[70]. TDIST

用途：返回学生氏 t 分布的百分点(概率)，t 分布中的数值(x)是 t 的计算值(将计算其百分点)。t 分布用于小样本数据集合的假设检验，使用此函数可以代替 t 分布的临界值表。

语法：TDIST(x, degrees_freedom, tails)

参数：X 为需要计算分布的数字，Degrees_freedom 为表示自由度的整数，Tails 指明返回的分布函数是单尾分布还是双尾分布。如果 tails=1，函数 TDIST 返回单尾分布。如果 tails=2，函数 TDIST 返回双尾分布。

实例：公式“=TDIST(60, 2, 1)”返回 0.000138831。

[71]. TINV

用途：返回作为概率和自由度函数的学生氏 t 分布的 t 值。

语法：TINV(probability, degrees_freedom)

参数：Probability 为对应于双尾学生氏-t 分布的概率，Degrees_freedom 为分布的自由度。

实例：公式“=TINV(0.5, 60)”返回 0.678600713。

[72]. TREND

用途：返回一条线性回归拟合线的一组纵坐标值(y 值)。即找到适合给定的数组 known_y's 和 known_x's 的直线(用最小二乘法)，并返回指定数组 new_x's 值在直线上对应的 y 值。

语法: TREND(known_y's, known_x's, new_x's, const)

参数: Known_y's 为已知关系 $y=mx+b$ 中的 y 值集合, Known_x's 为已知关系 $y=mx+b$ 中可选的 x 值的集合, New_x's 为需要函数 TREND 返回对应 y 值的新 x 值, Const 为逻辑值指明是否强制常数项 b 为 0。

[73]. TRIMMEAN

用途: 返回数据集的内部平均值。TRIMMEAN 函数先从数据集的头部和尾部除去一定百分比的数据点, 然后再求平均值。当希望在分析中剔除一部分数据的计算时, 可以使用此函数。

语法: TRIMMEAN(array, percent)

参数: Array 为需要进行筛选并求平均值的数组或数据区域, Percent 为计算时所要求除去的数据点的比例。如果 percent=0.2, 则在 20 个数据中除去 4 个, 即头部除去 2 个尾部除去 2 个。如果 percent=0.1, 30 个数据点的 10% 等于 3 个数据点。函数 TRIMMEAN 将对称地在数据集的头部和尾部各除去一个数据。

实例: 如果 A1=78、A2=45、A3=90、A4=12、A5=85, 则公式 “=TRIMMEAN(A1:A5, 0.1)” 返回 62。

[74]. TTEST

用途: 返回与学生氏-t 检验相关的概率。它可以判断两个样本是否来自两个具有相同均值的总体。

语法: TTEST(array1, array2, tails, type)

参数: Array1 是第一个数据集, Array2 是第二个数据集, Tails 指明分布曲线的尾数。如果 tails=1, TTEST 函数使用单尾分布。如果 tails=2, TTEST 函数使用双尾分布。Type 为 t 检验的类型。如果 type 等于 (1、2、3) 检验方法 (成对、等方差双样本检验、异方差双样本检验)。

实例: 公式 “=TTEST({3, 4, 5, 8, 9, 1, 2, 4, 5}, {6, 19, 3, 2, 14, 4, 5, 17, 1}, 2, 1)” 返回 0.196016。

[75]. VAR

用途: 估算样本方差。

语法: VAR(number1, number2, ...)

参数: Number1, number2, ... 对应于与总体样本的 1 到 30 个参数。

实例: 假设抽取某次考试中的 5 个分数, 并将其作为随机样本, 用 VAR 函数估算成绩方差, 样本值为 A1=78、A2=45、A3=90、A4=12、A5=85, 则公式 “=VAR(A1:A5)” 返回 1089.5。

[76]. VARA

用途: 用来估算给定样本的方差。它与 VAR 函数的区别在于文本和逻辑值(TRUE 和 FALSE)也将参与计算。

语法: VARA(value1, value2, ...)

参数: Value1, value2, ... 作为总体的一个样本的 1 到 30 个参数。

实例: 假设抽取某次考试中的 5 个分数, 并将其作为随机样本, 用 VAR 函数估算成

绩方差，样本值为 A1=78、A2=45、A3=90、A4=12、A5=85，则公式 “=VARA(A1:A5, TRUE)” 返回 1491.766667。

[77]. VARP

用途：计算样本总体的方差。

语法：VARP(number1, number2, ...)

参数：Number1, number2, ...为对应于样本总体的 1 到 30 个参数。其中的逻辑值 (TRUE 和 FALSE)和文本将被忽略。

实例：如果某次补考只有 5 名学生参加，成绩为 A1=88、A2=55、A3=90、A4=72、A5=85，用 VARP 函数估算成绩方差，则公式 “=VARP(A1:A5)” 返回 214.5。

[78]. VARPA

用途：计算样本总体的方差。它与 VARP 函数的区别在于文本和逻辑值(TRUE 和 FALSE)也将参与计算。

语法：VARPA(value1, value2, ...)

参数：Value1, value2, ...作为样本总体的 1 到 30 个参数。

实例：如果某次补考只有 5 名学生参加，成绩为 A1=88、A2=55、A3=90、A4=72、A5=85，用 VARPA 函数估算成绩方差，则公式 “=VARPA(A1:A5)” 返回 214.5。

[79]. WEIBULL

用途：返回韦伯分布。使用此函数可以进行可靠性分析，如设备的平均无故障时间

语法：WEIBULL(x, alpha, beta, cumulative)

参数：X 为用来计算函数值的数值，Alpha 分布参数，Beta 分布参数，Cumulative 指明函数的形式。

实例：公式 “=WEIBULL(98, 21, 100, TRUE)” 返回 0.480171231，=WEIBULL(58, 11, 67, FALSE)返回 0.031622583。

[80]. ZTEST

用途：返回 z 检验的双尾 P 值。Z 检验根据数据集或数组生成 x 的标准得分，并返回正态分布的双尾概率。可以使用此函数返回从某总体中抽取特定观测值的似然估计。

语法：ZTEST(array, x, sigma)

参数：Array 为用来检验 x 的数组或数据区域。X 为被检验的值。Sigma 为总体(已知)标准偏差，如果省略，则使用样本标准偏差。

实例：公式 “=ZTEST({3, 6, 7, 8, 6, 5, 4, 2, 1, 9}, 4)” 返回 0.090574。

11.文本和数据函数

[1]. ASC

用途：将字符串中的全角(双字节)英文字母更改为半角(单字节)字符。

语法：ASC(text)

参数: Text 为文本或包含文本的单元格引用。如果文本中不包含任何全角英文字母, 则文本不会被更改。

实例: 如果 A1=e x c e l , 则公式 “=ASC(A1)” 返回 excel。

[2]. CHAR

用途: 返回对应于数字代码的字符, 该函数可将其他类型的电脑文件中的代码转换为字符(操作环境为 MacintoshMacintosh 字符集和 WindowsANSI 字符集)。

语法: CHAR(number)

参数: Number 是用于转换的字符代码, 介于 1~255 之间(使用当前计算机字符集中的字符)。

实例: 公式 “=CHAR(56)” 返回 8, =CHAR(36)返回\$。

[3]. CLEAN

用途: 删除文本中不能打印的字符。对从其他应用程序中输入的字符串使用 CLEAN 函数, 将删除其中含有的当前操作系统无法打印的字符。

语法: CLEAN(text)

参数: Text 为要从中删除不能打印字符的任何字符串。

实例: 由于 CHAR(7) 返回一个不能打印的字符, 因此公式 “=CLEAN(CHAR(7)&"text"&CHAR(7))” 返回 text。

[4]. CODE

用途: 返回文字串中第一个字符的数字代码(对应于计算机当前使用的字符集)。

语法: CODE(text)

参数: Text 为需要得到其第一个字符代码的文本。

实例: 因为 CHAR(65)返回 A, 所以公式 “=CODE("Alphabet")” 返回 65。

[5]. CONCATENATE

用途: 将若干文字串合并到一个文字串中, 其功能与"&"运算符相同。

语法: CONCATENATE(text1, text2, ...)

参数: Text1, text2, ...为 1 到 30 个将要合并成单个文本的文本项, 这些文本项可以是文字串、数字或对单个单元格的引用。

实例: 如果 A1=98、A2=千米, 则公式 “=CONCATENATE(A1, A2)” 返回 “98 千米”, 与公式 “=A1&A2” 等价。

[6]. DOLLAR 或 RMB

用途: 按照货币格式将小数四舍五入到指定的位数并转换成文字。

语法: DOLLAR(number, decimals)或 RMB(number, decimals)

参数: Number 是数字、包含数字的单元格引用, 或计算结果为数字的公式; Decimals 是十进制的小数, 如果 Decimals 为负数, 则参数 number 从小数点往左按相应位数取整。如果省略 Decimals, 则假设其值为 2。

实例: 公式 “=RMB(1586.567, 2)” 返回 “¥ 1586.57”, =RMB(99.888) 返回

“¥99.89”。

[7]. EXACT

用途：测试两个字符串是否完全相同。如果它们完全相同，则返回 TRUE；否则返回 FALSE。EXACT 函数能区分大小写，但忽略格式上的差异。

语法：EXACT(text1, text2)

参数：Text1 是待比较的第一个字符串，Text2 是待比较的第二个字符串。

实例：如果 A1=物理、A2=化学 A3=物理，则公式 “=EXACT(A1, A2)” 返回 FALSE，=EXACT(A1, A3)返回 FALSE，=EXACT("word", "word")返回 TRUE。

[8]. FIND

用途：FIND 用于查找其他文本串 (within_text) 内的文本串 (find_text)，并从 within_text 的首字符开始返回 find_text 的起始位置编号。此函数适用于双字节字符，它区分大小写但不允许使用通配符。

语法：FIND(find_text, within_text, start_num)

参数：Find_text 是待查找的目标文本；Within_text 是包含待查找文本的源文本；Start_num 指定从其开始进行查找的字符，即 within_text 中编号为 1 的字符。如果忽略 start_num，则假设其为 1。

实例：如果 A1=软件报，则公式 “=FIND("软件", A1, 1)” 返回 1。

[9]. FINDB

用途：FINDB 用于查找其他文本串 (within_text) 内的文本串 (find_text)，并根据每个字符使用的字节数从 within_text 的首字符开始返回 find_text 的起始位置编号。FIND 与 FINDB 的区别在于：前者是以字符数为单位返回起始位置编号，后者是以字节数为单位返回起始位置编号。

语法：FINDB(find_text, within_text, start_num)

参数：Find_text 是待查找的目标文本；Within_text 是包含待查找文本的源文本；Start_num 指定从其开始进行查找的字符，即 within_text 中编号为 1 的字符。如果忽略 start_num，则假设其为 1。注意：此函数适用于双字节字符，它能区分大小写但不允许使用通配符。其它事项与 FIND 函数相同。

实例：如果 A1=电脑爱好者，则公式 “=FINDB(爱好者", A1, 1)” 返回 5。因为每个字符均按字节进行计算，而一个汉字为 2 个字节，所以第三个汉字“爱”从第五个字节开始。

[10]. FIXED

用途：按指定的小数位数四舍五入一个数，以小数格式设置后以文字串形式返回结果。

语法：FIXED(number, decimals, no_commas)

参数：Number 是要进行四舍五入并转换成文字串的数；Decimals 为一数值，用以指定小数点右边的小数位数；No_commas 为一逻辑值。如果是 TRUE，则函数 FIXED 返回的文字不含逗号。如果 no_commas 是 FALSE 或省略，则返回的文字中包含逗号。

实例：如果 A1=2001.16845，则公式 “=FIXED(A2, 2, TRUE)” 返回 2001.17，=FIXED(6834.567, -1)返回 6830。

[11]. JIS

用途：将字符串中的半角(单字节)英文字母更改为全角(双字节)字符。

语法：JIS(text)

参数：Text 为文本或对包含文本的单元格(或引用)。如果文本中不包含任何半角英文字母，则文本不会更改。

实例：如果 A1=excel，则公式 “=JIS(a1)” 返回 EXCEL。

[12]. LEFT 或 LEFTB

用途：根据指定的字符数返回文本串中的第一个或前几个字符。此函数用于双字节字符。

语法：LEFT(text, num_chars)或 LEFTB(text, num_bytes)

参数：Text 是包含要提取字符的文本串；Num_chars 指定函数要提取的字符数，它必须大于或等于 0。Num_bytes 按字节数指定由 LEFTB 提取的字符数。

实例：如果 A1=电脑爱好者，则 LEFT(A1, 2)返回 “电脑”，LEFTB(A1, 2)返回 “电”。

[13]. LEN 或 LENB

用途：LEN 返回文本串的字符数。LENB 返回文本串中所有字符的字节数。

语法：LEN(text)或 LENB(text)。

参数：Text 待要查找其长度的文本。注意：此函数用于双字节字符，且空格也将作为字符进行统计。

实例：如果 A1=电脑爱好者，则公式 “=LEN(A1)” 返回 5，=LENB(A1)返回 10。

[14]. LOWER

用途：将一个文字串中的所有大写字母转换为小写字母。

语法：LOWER(text)

语法：Text 是包含待转换字母的文字串。注意：LOWER 函数不改变文字串中非字母的字符。LOWER 与 PROPER 和 UPPER 函数非常相似。

实例：如果 A1=Excel，则公式 “=LOWER(A1)” 返回 excel。

[15]. MID 或 MIDB

用途：MID 返回文本串中从指定位置开始的特定数目的字符，该数目由用户指定。MIDB 返回文本串中从指定位置开始的特定数目的字符，该数目由用户指定。MIDB 函数可以用于双字节字符。

语法：MID(text, start_num, num_chars)或 MIDB(text, start_num, num_bytes)

参数：Text 是包含要提取字符的文本串。Start_num 是文本中要提取的第一个字符的位置，文本中第一个字符的 start_num 为 1，以此类推；Num_chars 指定希望 MID 从文本中返回字符的个数；Num_bytes 指定希望 MIDB 从文本中按字节返回字符的个数。

实例：如果 a1=电子计算机，则公式 “=MID(A1, 3, 2)” 返回 “计算”，=MIDB(A1, 3, 2)返回 “子”。

[16]. PHONETIC

用途：提取文本串中的拼音(furigana)字符。

语法：PHONETIC(reference)

参数：Reference 为文本串或对单个单元格或包含文本串的单元格区域的引用。如果 reference 为单元格区域，则返回区域左上角单元格中的 furigana 文本串。如果 reference 为不相邻单元格的区域，将返回#N/A 错误值。注意：该函数在中文 Excel 中无效。

[17]. PROPER

用途：将文字串的首字母及任何非字母字符之后的首字母转换成大写。将其余的字母转换成小写。

语法：PROPER(text)

参数：Text 是需要进行转换的字符串，包括双引号中的文字串、返回文本值的公式或对含有文本的单元格引用等。

实例：如果 A1=学习 excel，则公式 “=PROPER(A1)” 返回 “学习 Excel”。

[18]. REPLACE 或 REPLACEB

用途：REPLACE 使用其他文本串并根据所指定的字符数替换另一文本串中的部分文本。REPLACEB 的用途与 REPLACE 相同，它是根据所指定的字节数替换另一文本串中的部分文本。

语法：REPLACE(old_text,start_num,num_chars,new_text), REPLACEB(old_text,start_num,num_bytes,new_text)

参数：Old_text 是要替换其部分字符的文本；Start_num 是要用 new_text 替换的 old_text 中字符的位置；Num_chars 是希望 REPLACE 使用 new_text 替换 old_text 中字符的个数；Num_bytes 是希望 REPLACE 使用 new_text 替换 old_text 的字节数；New_text 是要用于替换 old_text 中字符的文本。注意：以上两函数均适用于双字节的汉字。

实例：如果 A1=学习的革命、A2=电脑，则公式 “=REPLACE(A1, 3, 3, A2)” 返回 “学习电脑”，=REPLACEB(A1, 2, 3, A2)返回 “电脑的革命”。

[19]. REPT

用途：按照给定的次数重复显示文本。可以通过 REPT 函数对单元格进行重复填充。

语法：REPT(text, number_times)

参数：Text 是需要重复显示的文本，Number_times 是重复显示的次数(正数)。注意：REPT 函数的结果不能多于 255 个字符。

实例：公式 “=REPT(“软件报”, 2)” 返回 “软件报软件报”。

[20]. RIGHT 或 RIGHTB

用途：RIGHT 根据所指定的字符数返回文本串中最后一个或多个字符。RIGHTB 根据所指定的字节数返回文本串中最后一个或多个字符。

语法: RIGHT(text, num_chars), RIGHTB(text, num_bytes)

参数: Text 是包含要提取字符的文本串; Num_chars 指定希望 RIGHT 提取的字符数, 它必须大于或等于 0。如果 num_chars 大于文本长度, 则 RIGHT 返回所有文本。如果忽略 num_chars, 则假定其为 1。Num_bytes 指定欲提取字符的字节数。

实例: 如果 A1=学习的革命, 则公式 “=RIGHT(A1,2)” 返回 “革命”, =RIGHTB(A1, 2)返回 “命”。

[21]. SEARCH 或 SEARCHB

用途: 返回从 start_num 开始首次找到特定字符或文本串的位置编号。其中 SEARCH 以字符数为单位, SEARCHB 以字节数为单位。

语法: SEARCH(find_text, within_text, start_num), SEARCHB(find_text, within_text, start_num)

参数: Find_text 是要查找的文本, 可以使用通配符, 包括问号 “?” 和星号 “*”。其中问号可匹配任意的单个字符, 星号可匹配任意的连续字符。如果要查找实际的问号或星号, 应当在该字符前键入波浪线 “~”。Within_text 是要在其中查找 find_text 的文本。Start_num 是 within_text 中开始查找的字符的编号。如果忽略 start_num, 则假定其为 1。

实例: 如果 A1=学习的革命, 则公式 “=SEARCH("的", A1)” 返回 3, =SEARCHB("的", A1)返回 5。

[22]. SUBSTITUTE

用途: 在文字串中用 new_text 替代 old_text。如果需要在在一个文字串中替换指定的文本, 可以使用函数 SUBSTITUTE; 如果需要在某一文字串中替换指定位置处的任意文本, 就应当使用函数 REPLACE。

语法: SUBSTITUTE(text, old_text, new_text, instance_num)

参数: Text 是需要替换其中字符的文本, 或是含有文本的单元格引用; Old_text 是需要替换的旧文本; New_text 用于替换 old_text 的文本; Instance_num 为一数值, 用来指定以 new_text 替换第几次出现的 old_text; 如果指定了 instance_num, 则只有满足要求的 old_text 被替换; 否则将用 new_text 替换 Text 中出现的所有 old_text。

实例: 如果 A1=学习的革命、A2=电脑, 则公式 “=SUBSTITUTE(A1, "的革命", A2, 1)” 返回 “学习电脑”。

[23]. T

用途: 将数值转换成文本。

语法: T(value)

参数: Value 是需要进行测试的数据。如果 value 本身就是文本, 或是对文本单元格的引用, T 函数将返回 value; 如果没有引用文本, 则返回 ""(空文本)。

实例: 如果 A1 中含有文本 “电脑”, 则公式 “=T(A1)” 返回 “电脑”。

[24]. TEXT

用途: 将数值转换为按指定数字格式表示的文本。

语法: TEXT(value, format_text)

参数: Value 是数值、计算结果是数值的公式、或对数值单元格的引用; Format_text

是所要选用的文本型数字格式，即“单元格格式”对话框“数字”选项卡的“分类”列表框中显示的格式，它不能包含星号“*”。注意：使用“单元格格式”对话框的“数字”选项卡设置单元格格式，只会改变单元格的格式而不会影响其中的数值。使用函数 TEXT 可以将数值转换为带格式的文本，而其结果将不再作为数字参与计算。

实例：如果 A1=2986.638，则公式 “=TEXT(A5, “#，##0.00”)” 返回 2, 986.64。

[25]. TRIM

用途：除了单词之间的单个空格外，清除文本中的所有的空格。如果从其他应用程序中获得了带有不规则空格的文本，可以使用 TRIM 函数清除这些空格。

语法：TRIM(text)

参数：Text 是需要清除其中空格的文本。

实例：如果 A1=FirstQuarterEarnings，则公式 “=TRIM(A1)” 返回 “FirstQuarterEarnings”。

[26]. UPPER

用途：将文本转换成大写形式。

语法：UPPER(text)

参数：Text 为需要转换成大写形式的文本，它可以是引用或文字串。

实例：公式 “=UPPER(“apple”)” 返回 APPLE。

[27]. VALUE

用途：将表示数字的文字串转换成数字。

语法：VALUE(text)

参数：Text 为带引号的文本，或对需要进行文本转换的单元格的引用。它可以是 Excel 可以识别的任意常数、日期或时间格式。如果 Text 不属于上述格式，则 VALUE 函数返回错误值 #VALUE!。注意：通常不需要在公式中使用 VALUE 函数，Excel 可以在需要时自动进行转换。VALUE 函数主要用于与其他电子表格程序兼容。

实例：公式 “=VALUE(“¥ 1, 000”)” 返回 1000；
=VALUE(“16:48:00”)-VALUE(“12:00:00”)返回 0.2，该序列数等于 4 小时 48 分钟。

[28]. WIDECHAR

用途：将单字节字符转换为双字节字符。

语法：WIDECHAR(text)

参数：Text 是需要转换为双字节字符的文本或包含文本的单元格引用。注意：因为汉字本身是双字节字符，所以使用此函数转换汉字时得到的是汉字的原形。

实例：公式 “=WIDECHAR(“apple”)” 返回 apple，=WIDECHAR(“电脑”)返回 “电脑”。

三. 函数应用案例——算账理财

1. 零存整取储蓄

“零存整取”是工薪阶层常用的投资方式，这就需要计算该项投资的未来值，从而决定是否选择某种储蓄方式。

[1]. 函数分解

FV 函数基于固定利率及等额分期付款方式，返回某项投资的未来值。

语法：FV(rate,nper,pmt,pv,type)Rate 为各期利率；Nper 为总投资期，即该项投资的付款期总数；Pmt 为各期所应支付的金额，其数值在整个年金期间保持不变；Pv 为现值，即从该项投资开始计算时已经入账的款项，或一系列未来付款的当前值的累积和；Type 为数字 0 或 1，用以指定各期的付款时间是在期初还是期末。

[2]. 实例分析

新建一个工作表，在其 A1、B1、C1、D1 单元格分别输入“投资利率”、“投资期限”、“投资金额”和“账户初始金额”。假设妻子新建一个账户每月底存入 300 元，年利 2.1%(即月息 0.00175)，连续存款 5 年，可以在 A2、B2、C2、D2 单元格分别输入“0.00175”、“60”、“500”和“1”。然后选中 E2 单元格输入公式“=FV(A2,B2,-C2,D2,1)”，回车即可获得该投资的到期本金合计为“¥18,994.67”。公式中的“-C2”表示资金是支出的，“C2”前不加负号也可，这样计算出来的结果就是负值。如果丈夫也有“零存整取”账户，每月初存入 200 元，年利 1.28%(即月息 0.001667)，连续存款 3 年，可以在 A3、B3、C3、D3 单元格分别输入“0.001667”、“36”、“200”和“0”。然后把 E2 单元格中的公式复制到 E3 单元格(将光标指向 E2 单元格的拖动柄，当黑色十字光标出现后向下拖动一格)，即可得知该投资的到期本金合计“¥7,426.42”。提示：上述计算结果包括本金和利息，但不包括利息税等其他费用。

2. 还贷金额

如今贷款购买住房进行消费的家庭越来越多，计算贷款的月偿还金额是决策的重要依据，下面我们就来设计如何知道自己每月的还款金额。

[1]. 函数分解

PMT 函数基于固定利率及等额分期付款方式，返回贷款的每期付款额。

语法：PMT(rate,nper,pv,fv,type)Rate 为贷款利率；Nper 为该项贷款的付款总数；Pv 为现值，或一系列未来付款的当前值的累积和；Fv 为未来值，或在最后一次付款后希望得到的现金余额；Type 为数字 0 或 1。

[2]. 实例分析

新建一个工作表，在其 A1、B1、C1、D1 单元格分别输入“贷款利率”、“还贷年

限”、“贷款金额”和“还贷时间”。

假设贷款年利为 4.1%(即月息 0.00342), 预计的还贷时间为 10 年, 贷款金额为 10 万元, 且每月底还贷。可以在 A2、B2、C2、D2 单元格分别输入“0.00342”、“360”、“100000”和“1”(表示月末还贷, 0 表示月初还贷)。然后选中 E2 单元格输入公式“=PMT(A2,B2,C2,,D2)”, 回车就可以获得每月的还款金额为“¥-481.78”。上式中 C2 后的两个逗号之间还有一个参数, 表示还贷期限结束时账户上的余额, 对这个例子来说应该是 0, 所以可以忽略该参数或写成“=PMT(A2,B2,C2,0,D2)”。

3. 保险收益

保险公司开办了一种平安保险, 具体办法是一次性缴费 12000 元, 保险期限为 20 年。如果保险期限内没有出险, 每年返还 1000 元。请问在没有出险的情况下, 它与现在的银行利率相比, 这种保险的收益率如何。

[1]. 函数分解

RATE 函数返回投资的各期利率。该函数通过迭代法计算得出, 并且可能无解或有多个解。

语法: RATE(nper,pmt,pv,fv,type,guess) Nper 为总投资期, 即该项投资的付款期总数; Pmt 为各期付款额, 其数值在整个投资期内保持不变; Pv 为现值, 即从该项投资开始计算时已经入帐的款项, 或一系列未来付款当前值的累积和; Fv 为未来值, 或在最后一次付款后希望得到的现金余额; Type 为数字 0 或 1。

[2]. 实例分析

新建一个工作表, 在其 A1、B1、C1、D1 单元格分别输入“保险年限”、“年返还金额”、“保险金额”、“年底返还”和“现行利息”。然后在 A2、B2、C2、D2 和 E2 单元格分别输入“20”、“1000”、“12000”、“1”(表示年底返还, 0 表示年初返还)和“0.02”。然后选中 F2 单元格输入公式“=RATE(A2,B2,C2,,D2,E2)”, 回车就可以获得该保险的年收益率为“0.06”。要高于现行的银行存款利率, 所以还是有利可图的。上面公式中的 C2 后面有两个逗号, 说明最后一次付款后账面上的现金余额为零。

4. 个税缴纳金额

假设个人收入调节税的收缴标准是:

工资在 800 元以下的免征调节税, 工资 800 元以上至 1500 元的超过部分按 5%的税率征收, 1500 元以上至 2000 元的超过部分按 8%的税率征收, 高于 2000 元的超过部分按 20%的税率征收。

我们可以按以下方法设计一个可以修改收缴标准的工作簿: 新建一个工作表, 在其 A1、B1、C1、D1、E1 单元格分别输入“姓名”、“工资总额”、“扣款”、“个税”和“实付工资”。

为了方便个税标准的修改, 我们可以另外打开一个工作表(例如 Sheet2), 在其 A1、B1、C1、D1、E1 单元格中输入“免征标准”、“低标准”、“中等标准”和“高标准”, 然后分别在其下方的单元格内输入“800”、“1500”、“2000”、“2000”。接下来回到工作表 Sheet1 中, 选中 D 列的 D2 单元格输入公式

“=IF(C2<=Sheet2!A2,"",IF((C2-Sheet2!A2)<=Sheet2!B2,(C2-Sheet2!A2)*0.05,IF(C2-Sheet2!C2<=Sheet2!C2,(C2-Sheet2!C2)*0.08,IF(C2>Sheet2!D2,(C2-Sheet2!D2)*0.2))))”,回车后即可计算出 C2 单元格中的应缴个税金额。此后用户只需把公式复制到 C3、C4 等单元格，就可以计算出其他职工应缴纳的个税金额。

上述公式的特点是把个税的征收标准放到另一个工作表中，如果征税标准发生了变化，用户只需修改相应单元格中的数值，不需要对公式进行修改，可以减少发生计算错误的可能。公式中的 IF 语句是逐次计算的，如果第一个逻辑判断 “C2-Sheet2!A2)<=Sheet2!B2” 成立，即工资收入低于征收标准，则个税计算公式所在单元格被填入空格；如果第一个逻辑判断式不成立，则计算第二个 IF 语句，直至计算结束。假如征税标准多于 4 个，可以按上述继续嵌套 IF 函数(最多 7 个)。

四. 函数应用案例——信息统计

使用 Excel 管理人事信息，具有无须编程、简便易行的特点。假设有一个人事管理工作表，它的 A1、B1、C1、D1、E1、F1、G1 和 H1 单元格分别输入“序号”、“姓名”、“身份证号码”、“性别”、“出生年月”等。自第 2 行开始依次输入职工的人事信息。为了尽可能减少数据录入的工作量，下面利用 Excel 函数实现数据统计的自动化。

1. 性别输入

根据现行的居民身份证号码编码规定，正在使用的 18 位的身份证编码。它的第 17 位为性别(奇数为男，偶数为女)，第 18 位为效验位。而早期使用的是 15 位的身份证编码，它的第 15 位是性别(奇数为男，偶数为女)。

[1]. 函数分解

LEN 函数返回文本字符串中的字符数。

语法：LEN(text)Text 是要查找其长度的文本。空格将作为字符进行计数。

MOD 函数返回两数相除的余数。结果的正负号与除数相同

语法：MOD(number,divisor)Number 为被除数；Divisor 为除数。MID 函数返回文本字符串中从指定位置开始的特定数目的字符，该数目由用户指定。

语法：MID(text,start_num,num_chars)Text 为包含要提取字符的文本字符串；Start_num 为文本中要提取的第一个字符的位置。文本中第一个字符的 start_num 为 1，以此类推；Num_chars 指定希望 MID 从文本中返回字符的个数。

[2]. 实例分析

为了适应上述情况，必须设计一个能够适应两种身份编码的性别计算公式，在 D2 单元格中输入 “=IF(LEN(C2)=15,IF(MOD(MID(C2,15,1),2)=1,"男","女"),IF(MOD(MID(C2,17,1),2)=1,"男","女"))”。回车后即可

在单元格获得该职工的性别，而后只要把公式复制到 D3、D4 等单元格，即可得到其他职工的性别。为了便于大家了解上述公式的设计思路，下面简单介绍一下它的工作原理：该公式由三个 IF 函数构成，其中 “IF(MOD(MID(C2,15,1),2)=1,"男","女")” 和 “IF(MOD(MID(C2,17,1),2)=1,"男","女")” 作为第一个函数的参数。公式中 “LEN(C2)=15”

是一个逻辑判断语句，LEN 函数提取 C2 等单元格中的字符长度，如果该字符的长度等于 15，则执行参数中的第一个 IF 函数，否则就执行第二个 IF 函数。在参数“IF(MOD(MID(C2,15,1),2)=1,"男","女")”中，MID 函数从 C2 的指定位置(第 15 位)提取 1 个字符，而 MOD 函数将该字符与 2 相除，获取两者的余数。如果两者能够除尽，说明提取出来的字符是 0(否则就是 1)。逻辑条件“MOD(MID(C2,15,1),2)=1”不成立，这时就会在 D2 单元格中填入“女”，反之则会填入“男”。如果 LEN 函数提取的 C2 等单元格中的字符长度不等于 15，则会执行第 2 个 IF 函数。除了 MID 函数从 C2 的指定位置(第 17 位，即倒数第 2 位)提取 1 个字符以外，其他运算过程与上面的介绍相同。

2. 出生日期输入

[1]. 函数分解

CONCATENATE 函数将几个文本字符串合并为一个文本字符串。

语法：CONCATENATE(text1,text2,...)Text1,text2,...为 1~30 个要合并成单个文本项的文本项。文本项可以为文本字符串、数字或对单个单元格的引用。

[2]. 实例分析

与上面的思路相同，我们可以在 E2 单元格中输入公式“=IF(LEN(C2)=15,CONCATENATE("19",MID(C2,7,2)," 年 ",MID(C2,9,2)," 月 ",MID(C2,11,2)," 日 "),CONCCTENCTE(MID(C2,7,4)," 年 ",MID(C2,11,2)," 月 ",MID(C2,13,2),"日"))”。其中“LEN(C2)=15”仍然作为逻辑判断语句使用，它可以判断身份证号码是 15 位的还是 18 位的，从而调用相应的计算语句。对 15 位的身份证号码来说，左起第 7 至 12 个字符表示出生年、月、日，此时可以使用 MID 函数从身份证号码的特定位置，分别提取出生年、月、日。然后用 CONCATENATE 函数将提取出来的文字合并起来，就能得到对应的出生年月日。公式中“19”是针对早期身份证号码中存在 2000 年问题设计的，它可以在计算出来的出生年份前加上“19”。对“18”位的身份证号码的计算思路相同，只是它不存在 2000 年问题，公式中不用给计算出来的出生年份前加上“19”。注意：CONCATENATE 函数和 MID 函数的操作对象均为文本，所以存放身份证号码的单元格必须事先设为文本格式，然后再输入身份证号。

3. 职工信息查询

Excel 提供的“记录单”功能可以查询记录，如果要查询人事管理工作表中的某条记录，然后把它打印出来，必须采用下面介绍的方法。

[1]. 函数分解

INDEX 函数返回数据清单或数组中的元素值，此元素由行序号和列序号的索引值给定。INDEX 函数有两种语法形式：数组和引用。数组形式通常返回数值或数值数组，引用形式通常返回引用。当函数 INDEX 的第一个参数为数组常数时，使用数组形式。

语法 1(数组形式)：INDEX(array,row_num,column_num)，Array 为单元格区域或数组常量。如果数组只包含一行或一列，则相对应的参数 row_num 或 column_num 为可选。如果数组有多行和多列，但只使用 row_num 或 column_num，函数 INDEX 返回

数组中的整行或整列，且返回值也为数组；Row_num 为数组中某行的行序号，函数从该行返回数值。如果省略 row_num，则必须有 column_num；Column_num 为数组中某列的列序号，函数从该列返回数值。如果省略 column_num，则必须有 row_num。

语法 2(引用形式)：INDEX(reference,row_num,column_num,area_num)，Reference 表示对一个或多个单元格区域的引用。如果为引用输入一个不连续的区域，必须用括号括起来。如果引用中的每个区域只包含一行或一列，则相应的参数 row_num 或 column_num 分别为可选项；Row_num 引用中某行的行序号，函数从该行返回一个引用；Column_num 引用中某列的列序号，函数从该列返回一个引用；Area_num 选择引用中的一个区域，并返回该区域中 row_num 和 column_num 的交叉区域。选中或输入的的第一个区域序号为 1，第二个为 2，以此类推。如果省略 area_num，函数 INDEX 使用区域 1。MATCH 函数返回在指定方式下与指定数值匹配的数组中元素的相应位置。语法：MATCH(lookup_value,lookup_array,match_type)Lookup_value 为需要在数据表中查找的数值；Lookup_value 为需要在 Look_array 中查找的数值；Match_type 为数字 -1、0 或 1。

[2]. 实例分析

如果上面的人事管理工作表放在 Sheet1 中，为了防止因查询操作而破坏它(必要时可以添加只读保护)，我们可以打开另外一个空白工作表 Sheet2，把上一个数据清单中的列标记复制到第一行。假如你要以“身份证号码”作为查询关键字，就要在 C2 单元格中输入公式“=INDEX(Sheet1!C2:C600,MATCH(SCS5,Sheet1!SCS2:SCS600,0),1)”。其中的参数“SCS5”引用公式所在工作表中的 C5 单元格(也可以选用其他单元格)，执行查询时要在其中输入查询关键字，也就是待查询记录中的身份证号码。参数“Sheet1!C2:C600”设定 INDEX 函数的查询范围，引用的是数据清单 C 列的所有单元格。MATCH 函数中的参数“0”指定它查找“Sheet1!SCS2:SCS600”区域中等于 SCS5 的第一个值，并且引用的区域“Sheet1!SCS2:SCS600,0”可以按任意顺序排列。上面的公式执行数据查询操作时，首先由 MATCH 函数在“Sheet1!SCS2:SCS600”区域搜索，找到“SCS5”单元格中的数据在引用区域中的位置(自上而下第几个单元格)，从而得知待查询数据在引用区域中的第几行。接下来 INDEX 函数根据 MATCH 函数给出的行号，返回“Sheet1!C2:C600”区域中对应行数单元格中的数据。假设其中待查询的“身份证号码”是“3234567896”，它位于“Sheet1!SCS2:SCS600”区域的第三行，MATCH 函数就会返回“3”。接着 INDEX 函数返回“Sheet1!C2:C600”区域中行数是“3”的数据，也就是“3234567896”。然后，我们将光标放到 C2 单元格的填充柄上，当十字光标出现以后向右拖动，从而把 C2 中的公式复制到 D2、E2 等单元格(然后再向左拖动，以便把公式复制到 B2、A2 单元格)，这样就可以获得与该身份证号对应的性别、籍贯等数据。注意：公式复制到 D2、E2 等单元格以后，INDEX 函数引用的区域就会发生变化，由 C2:C600 变成 D2:D600、E2:E600 等等。但是 MATCH 函数返回的(相对)行号仍然由查询关键字给出，此后 INDEX 函数就会根据 MATCH 函数返回的行号从引用区域中找到数据。在 Sheet2 工作表中进行查询时只要在查询输入单元格中输入关键字，回车后即可在工作表的 C2 单元格内看到查询出来的身份证号码。如果输入的身份证号码关键字不存在或输入错误，则单元格内会显示“#N/A”字样。

4. 职工性别统计

[1]. 函数分解

COUNTIF 函数计算区域中满足给定条件的单元格的个数。

语法: COUNTIF(range,criteria) Range 为需要计算其中满足条件的单元格数目的单元格区域; Criteria 为确定哪些单元格将被计算在内的条件, 其形式可以为数字、表达式或文本。

[2]. 实例分析

假设上面使用的人事管理工作表中有 599 条记录, 统计职工中男性和女性人数的方法是: 选中单元格 D601(或其他用不上的空白单元格), 统计男性职工人数可以在其中输入公式 “=男”&COUNTIF(D2:D600,"男")&"人"; 接着选中单元格 D602, 在其中输入公式 “=女”&COUNTIF(D2:D227,"女")&"人”。回车后即可得到 “男 399 人”、“女 200 人”。

上式中 D2:D600 是对 “性别” 列数据区域的引用, 实际使用时必须根据数据个数进行修改。“男” 或 “女” 则是条件判断语句, 用来判断区域中符合条件的数据然后进行统计。

“&” 则是字符连接符, 可以在统计结果的前后加上 “男”、“人” 字样, 使其更具有可读性。

5. 年龄统计

在人事管理工作中, 统计分布在各个年龄段中的职工人数也是一项经常性工作。假设上面介绍的工作表的 E2:E600 单元格存放职工的工龄, 我们要以 5 年为一段分别统计年龄小于 20 岁、20 至 25 岁之间, 一直到 55 至 60 岁之间的年龄段人数, 可以采用下面的操作方法。

[1]. 函数分解

FREQUENCY 函数以一系列垂直数组返回某个区域中数据的频率分布。

语法: FREQUENCY(data_array,bins_array)

Data_array 为一数组或对一组数值的引用, 用来计算频率。如果 data_array 中不包含任何数值, 函数 FREQUENCY 返回零数组; Bins_array 为间隔的数组或对间隔的引用, 该间隔用于对 data_array 中的数值进行分组。如果 bins_array 中不包含任何数值, 函数 FREQUENCY 返回 data_array 中元素的个数。

[2]. 实例分析

首先在工作表中找到空白的 I 列(或其他列), 自 I2 单元格开始依次输入 20、25、30、35、40...60, 分别表示统计年龄小于 20、20 至 25 之间、25 至 30 之间等的人数。然后在 I2 列旁边选中相同个数的单元格, 例如 J2: J10 准备存放各年龄段的统计结果。然后在编辑栏输入公式 “=FREQUENCY(YEAR(TODAY())-YEAR(E2:E600),I2:I10)”, 按下 Ctrl+Shift+Enter 组合键即可在选中单元格中看到计算结果。其中位于 J2 单元格中的结果表示年龄小于 20 岁的职工人数, J3 单元格中的数值表示年龄在 20 至 25 之间的职工人数

等。

6. 名次值统计

在工资统计和成绩统计等场合，往往需要知道某一名次(如工资总额第第三)的员工的工资是多少。这种统计的操作方法如下。

[1]. 函数分解

LARGE 函数返回数据集中第 K 个最大值。使用此函数可以根据相对标准来选择数值。

语法：LARGE(array,k)

Array 为需要从中选择第 K 个最大值的数组或数据区域；K 为返回值在数组或数据单元格区域中的位置(从大到小排)。SMALL 函数返回数据集中第 K 个最小值。使用此函数可以返回数据集中特定位置上的数值。法：SMALL(array,k)Array 为需要找到第 K 个最小值的数组或数字型数据区域；K 为返回的数据在数组或数据区域里的位置(从小到大)。

[2]. 实例分析

假设 C2:C688 区域存放着员工的工资，首先在 D 列选取空白单元格 D3，在其中输入公式“=LARGE(C2:C688,D2)”。其中 D2 作为输入名次变量的单元格，如果你在其中输入 3，公式就可以返回 C2:C688 区域中第三大的数值。

如果我们把上述公式修改为“=SMALL(C2:C688,D1)”，然后在 D1 单元格中输入 6，就可以获得 C2:C688 区域倒数第六(小)的数值。为方便起见，你可以给 C2:C688 区域定义一个名称“职工工资”。此后可以把上述公式修改为“=LARGE(职工工资,D2)”或“=SMALL(职工工资,D1)”。

7. 位次阈值统计

与上例相似，在工资统计和成绩统计等场合，需要知道排名达到总体的前 1/3 的工资总额或分数(称为“阈值”)是多少。这种统计的操作方法如下：

[1]. 函数分解

PERCENTILE 函数返回区域中数值的第 K 个百分点的值。可以使用此函数来建立接受阈值。

语法：PERCENTILE(array,k)Array 为定义相对位置的数组或数据区域；K 为 0 到 1 之间的百分点值，包含 0 和 1。

[2]. 实例分析

假设 C2:C200 区域存放着学生的考试成绩，首先在 D 列选取空白单元格 D3，在其中输入公式“=PERCENTILE(C2:C200,D2)”。其中 D2 作为输入百分点变量的单元格，如果你在其中输入 0.33，公式就可以返回名次达到前 1/3 所需要的成绩。

五. 函数应用案例——管理计算

企业、学校等单位均存在许多管理计算问题，例如计算一个学期有几个授课日、企业在多少个工作日之后交货等等。下面介绍有关问题的几种计算方法。

1. 授课日数

[1]. 函数分解

NETWORKDAYS 函数专门用于计算两个日期值之间完整的工作日数值。这个工作日数值将不包括双休日和专门指定的其他各种假期。

语法: NETWORKDAYS(Start_date, End_date, Holidays) Start_date 表示开始日期; End_date 为终止日期, Holidays 表示作为特定假日的一个或多个日期。这些参数值既可以手工输入, 也可以对单元格的值进行引用。

[2]. 实例分析

假设新学期从 2003 年 9 月 1 日开始到 2004 年 1 月 15 日结束, 希望知道本学期有多少个授课日, 也就是排除双休日和国家法定假日外的授课工作日。这就是计算授课日数或工作日数的问题。首先打开一个空白工作表, 在 A1、B1、C1 单元格输入“开学时间”、“结束时间”、“法定节日”, 然后在其下面的单元格内输入“2003-9-1”、“2004-1-15”、“2003-10-1” “2003-10-2”、“2003-10-3”和“2004-1-1”(后四项必须在 C 列的“法定假日”下)。接着可以选中 D2 单元格, 输入公式“=NETWORKDAYS(A2,B2,C2:C5)”。公式中 A2 引用的是学期(或工作)的开始日期, B2 引用的是学期结束的日期, C2:C5 区域引用的是作为法定假日的多个日期。输入结束回车即可获得结果 95, 即 2003 年 9 月 1 日到 2004 年 1 月 15 日, 排除四个法定假日后的实际授课日是 95 天。

2. 折旧值计算

无论单位还是家庭, 许多固定资产和耐用消费品都存在折旧问题, 随着使用时间的延长, 其残值在不断减少。假设某单位有一批 2000 年购进原价 8500 元/每台的电脑, 预计使用寿命 6 年, 寿命期结束时的资产残值约为 1000 元, 要求使用第二年内的折旧值。

[1]. 函数分解

DB 函数使用固定余额递减法, 计算一笔资产在给定期间的折旧值。

语法: DB(cost, salvage, life, period, month)

Cost 为资产原值; Salvage 为资产在折旧期末的价值(也称为资产残值); Life 为折旧期限(有时也称作资产的使用寿命); Period 为需要计算折旧值的期间。Period 必须使用与 life 相同的单位; Month 为第一年的月份数, 如省略, 则假设为 12。

[2]. 实例分析

为了在参数改变以后仍能进行计算，我们打开一个空白工作表，在 A1、B1、C1、D1、E1 单元格输入“电脑原值”、“资产残值”、“使用寿命”、“折旧时间”和“折旧值”，然后在其下面的单元格内输入“8500”、“1000”、“6”、“2”。然后选中 E2 单元格在其中输入公式“=DB(A2,B2,C2,D2)”，回车后即可得到结果“¥ 1,785.00”，就是说使用期第二年的折旧值为 1785 元。如果你要计算其他设备或财产的折旧值，只需改变 A2、B2、C2、D2 单元格内的数值即可。

3. 客流均衡度计算

假设某超市周一到周六的客流人数是 16359、17254、18654、15398、21689 和 220867，总经理需要知道这种情况下的客流分布是否平坦。可以按如下方法计算：

[1]. 函数分解

KURT 函数返回数据集的峰值。峰值反映与正态分布相比某一分布的尖锐度或平坦度。正峰值表示相对尖锐的分布。负峰值表示相对平坦的分布。

语法：KURT(number1,number2,...)Number1,number2,...是用于计算峰值的 1~30 个参数。也可以不使用这种用逗号分隔参数的形式，而用单个数组或数组引用的形式。

[2]. 实例分析

打开一个空白工作表，在 A1 单元格中输入“一周客流统计”，然后将上述数据依次输入 A2、A3 等单元格。然后选中 A8 单元格，在其中输入公式“=KURT(D2:D7)”，回车即可获得结果“-1.719218897”，这说明超市的客流分布与正态分布相比是相对平坦的。假如星期天搞特价促销，客流增加到了 50867 人，则计算结果就会变为“5.45379941”。说明超市的客流分布与正态分布相比比较尖锐了，特价促销对客流的影响还是非常大的。需要指出的是，KURT 函数在教育统计等领域也有广泛用途，假如把函数引用的区域修改为许多分数的集合，就可以知道考试成绩的分布是否尖锐或平坦。

4. 销售额预测

假设某超市周一到周日的日销售额分别为 13、17、16、15、19、21 和 22(万元)，总经理需要预测今后一周内的日销售额的最高值和最低值。可以按如下方法进行预测：

[3]. 函数分解

TREND 函数返回一条线性回归拟合线的值。即找到适合已知数组 known_y's 和 known_x's 的直线(用最小二乘法)，并返回指定数组 new_x's 在直线上对应的 y 值

语法：TREND(known_y's,known_x's,new_x's,const)Known_y's 是关系表达式 $y=mx+b$ 中已知的 y 值集合；Known_x's 是关系表达式 $y=mx+b$ 中已知的可选 x 值集合；New_x's 为需要函数 TREND 返回对应 y 值的新 x 值；Const 为一逻辑值，用于指定是否将常量 b 强制设为 0。

[1]. 实例分析

首先要打开一个空白工作表，在 A1 单元格中输入“日销售额”，然后将上述数据依次输入 A2、A3 至 A8 单元格。然后选中 B2 至 B8 区域，在 Excel 的编辑栏输入公式“=TREND(A2:A8)”，回车即可在 B2 至 B8 区域获得 7 个结果，其中最高销售额为 21.64 万元，最低销售额为 13.5 万元。与 KURT 函数一样，TREND 函数可以用于教育统计中的学生入学数的峰值和低谷，铁路运输领域的客流高峰和低谷等的预测。

5. 客流与营业额的相关分析

[1]. 函数分解

CORREL 函数返回单元格区域 array1 和 array2 之间的相关系数。使用相关系数可以确定两种属性之间的关系。

语法：CORREL(array1,array2) Array1 为第一组数值单元格区域；Array2 为第二组数值单元格区域。

[2]. 实例分析

假设一个超市要分析客流量与营业额是否相关。首先运行 Excel 打开一个空白工作表，在 A1 至 A31 单元格输入八月份的每日客流人数，然后在 B1 至 B31 输入八月份每日的营业额，再将上述两个区域的名称定义为“日客流人数”和“日营业额”。接下来就可以选中工作表中的某个空白单元(例如 B32)，作为存储运算结果的位置。在 Excel 的编辑栏输入公式“=CORREL(日客流人数,日营业额)”，回车后即可在公式所在单元格看到相关系数的计算结果。上式中 CORREL 函数返回“日客流人数”和“日营业额”两个数据集合的相关系数，实际应用中必须根据要分析的数据集合对引用区域进行修改。与其他计算不同，CORREL 函数计算出的相关系数必须进行分析，才能得出两个数值之间是否相关的结论。统计理论根据各种因素(如“日客流人数”和“日营业额”)相互影响的关系，把相关分为正相关、负相关和零相关三种类型。

所谓正相关就是两个因素的变化方向相同，即同时变大或变小，例如气温和冷饮销量就是正相关；负相关就是两个因素的变化方向相反，即一个变大(小)另一个变小(大)，例如气温上升和羽绒服销量就是负相关；零相关就是两个因素的变化方向无规律，即不存在相互之间影响的情况，例如学生的考试成绩和面粉的销量就是零相关。

为了帮助不太熟悉统计理论的用户掌握 CORREL 函数的使用，这里使用相关程度分析的理论修改公式“=CORREL(日客流人数,日营业额)”，使之成为下面这种形式，从而更加直观的给出两列数据相关程度的结论。“=IF((ABS(CORREL(日客流人数,日营业额)))<=0.3,"相关程度低",IF((ABS(CORREL(日客流人数,日营业额)))<=0.5,"相关程度一般",IF((ABS(CORREL(日客流人数,日营业额)))<=0.7,"相关程度较高",IF((ABS(CORREL(日客流人数,日营业额)))<=0.9,"相关程度高",IF((ABS(CORREL(日客流人数,日营业额)))<=1,"相关程度极高")))))))”公式中的“CORREL(日客流人数,日营业额)”部分还是计算日客流人数和日营业额两列数据的相关系数。

由于这里只需要了解相关程度，所以使用 ABS 函数返回相关系数的绝对值。整个公式中由左往右的下一个 IF 语句就是上一个 IF 语句的参数。例如第一个逻辑判断表达式“(ABS(CORREL(日客流人数,日营业额)))<=0.3”为“真”(成立)，则公式所在单元格就

会被填入“相关程度低”；如果第一个逻辑判断表达式“(ABS(CORREL(日客流人数,日营业额)))<=0.3”为“假”(不成立)，则计算第二个 IF 语句“IF((ABS(CORREL(日客流人数,日营业额)))<=0.5)”；以此类推直至计算结束。