200+ Excel Formulas and Functions

The go-to-guide to master Microsoft Excel's many capabilities

Prof. Michael McDonald

www.bpbonline.com

Copyright © 2023 BPB Online

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor BPB Online or its dealers and distributors, will be held liable for any damages caused or alleged to have been caused directly or indirectly by this book.

BPB Online has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, BPB Online cannot guarantee the accuracy of this information.

First published: 2023

Published by BPB Online
WeWork
119 Marylebone Road
London NW1 5PU

UK | UAE | INDIA | SINGAPORE

ISBN 978-93-55513-786

www.bpbonline.com

Dedicated to

Excel Users Everywhere

About the Author

Prof. Michael McDonald is a partner with Morning Investments, a boutique financial training firm. He holds a PhD in finance and his work has been published in major financial journals such as the Journal of Corporate Finance, the Journal of Fixed Income, the Pacific Basin Finance Journal, and the Journal of Behavioral Finance, and has been quoted in the Wall Street Journal, CNN, Nasdaq.com, Bloomberg, Reuters, and many other outlets. Michael's past work experience includes working as a bond trader at Wachovia/Wells Fargo, as a quant at AQR, and in institutional investor product development for Seeking Alpha.

About the Reviewers

- **Prof. Pranav Gosalia** has over 16 years of corporate and academic experience in teaching, administration, training, and placement. He holds an MBA from Nirma University, a PGDCA from C-DAC, and a bachelor's degree from Gujarat University. He has worked with the IT, aviation, and education sectors.

  Prof. Gosalia has been the recipient of "Best SIP Mentor" for eight consecutive times, "Best Teacher Award – 2018" and "The Best Teacher Award Institution Building – 2015" instituted by the ICFAI Business School Alumni Federation (IBSAF). Apart from teaching for over six years, he has also been the coordinator for Quicforce – the Moodle-based Learning Management System; Digital Marketing initiative for college digital assets, and the Summer Internship Program (SIP) at IBS Ahmedabad.

  He is data enthusiastic and interested in MS Excel, Tableau, Business Analytics, and Marketing Analytics. He has done many Student Development programs and delivered sessions on case study-based teaching during Faculty Development Programs. He has also delivered a few corporate trainings on MS Excel, Tableau, and Data visualization.

- **Naresh** is a seasoned Fund Accounting professional with great interest and expertise in Microsoft Excel. Naresh has a background in data analytics and has been using Excel for over a decade. His comprehensive knowledge of Excel's features/functions and ability to convey complex concepts in an accessible manner, make him an invaluable resource for readers seeking reliable recommendations on MS Excel-related literature.

Acknowledgement

There are a few people I want to thank for the continued and ongoing support they have given me during the writing of this book. First and foremost, I would like to thank my family for continuously encouraging me in writing the book — I could have never completed this book without their support.

I am grateful to the course and the companies which gave me support throughout the learning process of using Excel and building data analytics skills. Thank you for the all the hidden support provided.

My gratitude also goes to the team at BPB Publications for being supportive enough to provide me quite a long time to finish the book.

Preface

This book covers over 200 functions in Excel, from basic functions all users need to know, and specialized tools for particular industries, to advanced functions that require some deep technical knowledge. This book also introduces the applications of these function in the industry and will help readers understand some of the practical opportunities to use Excel to make themselves more productive and proficient in their careers.

Chapter 1: Getting Started with Excel – In this short overview chapter, we will cover what the reader can expect to learn in the book, what they should know about Excel before getting started, and how they should use the book to get the most out of it.

Chapter 2: The 10 Most Popular Functions in Excel – In this chapter, we will cover the 10 most popular functions in Excel, including the SUM function, IF function, VLOOKUP function, MATCH function and many more.

Chapter 3: Logical Functions in Excel – Logical functions are used for organizing data and creating formulas in Excel. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 4: Lookup and Reference Functions in Excel – Excel is often used as a tool for building complex data driven models especially in business. To do this, look up and reference functions are used to create efficient datasets in Excel. In this chapter, we will cover the most popular and useful formulas that you need to know, to be able to use Excel effectively.

Chapter 5: Math Functions in Excel – Excel is filled with mathematical functions that can make virtually any computation much easier. This is an important task for students and workers alike. In this chapter, we will cover the most popular and useful formulas that you need to know, to be able to use Excel effectively.

Chapter 6: Statistical Functions in Excel – Statistical analysis is a foundational cornerstone of modern business and Excel is a great tool to help with doing that analysis. While Excel Add-ins can add tremendous versatility to the software, even without them, Excel is still very useful in a variety of situations where statistics are called for. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 7: Text Functions in Excel – Excel can help us clean up and organize datasets effectively, and this is especially important when dealing with mixed or semi-structured datasets. However, Excel can only help if we can use text functions proficiently. In this chapter, we will cover the most popular and useful formulas that you need to know, to be able to use Excel effectively.

Chapter 8: Information Functions in Excel – Like text functions, information functions can help us to go through a large dataset and quickly clean it up to make it useable for our work. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 9: Financial Functions in Excel – In finance, Excel is by far the most important and widely used software package. Part of the reason for that is the software contains numerous functions that let us do financial calculations quickly and accurately. These calculations can then be embedded into more complex financial models. While building models is a subject for another book, the first thing you need to do is start by just understanding the formulas and what they do. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 10: Date and Time Functions in Excel – Dates and times are a persistent hassle to deal with in datasets; they look like regular numbers although they actually are not. As a result, it is important to be able to use special Excel functions that let us work with dates and times when building larger models and datasets. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 11: Database Functions in Excel – Excel is not the most powerful database software on the market – but it is widely used for database work because it is ubiquitous, simple to learn, and it performs well for a variety of small database tasks. In this role, it is important to understand what the database functions Excel includes. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 12: Cube Functions in Excel – Cube functions are an unusual group of functions with some specific niche uses. While you may not be familiar with them, they can be very important in specific settings. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Chapter 13: Web Functions in Excel – Excel does not have many of the more sophisticated tools of other software packages that interact and pull information from the web, but it does have some specific tools that you should be familiar with. In this chapter, we will cover the most popular and useful formulas that you need to know, to be able to use Excel effectively.

Chapter 14: User Defined Functions in Excel – User defined functions are advanced but very powerful functions that can enable Excel to greatly speed up tedious and time-consuming tasks. While these functions are harder to use, they can be a huge time-saver for people doing specific kinds of projects. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

This text will be useful to you in your journey to understand and improve your knowledge in Excel. The goal of this book is to be a desk reference that you can pull out and refer to, when you are unsure about a particular function and how to use it. Throughout this text, we will cover many functions that serve a variety of purposes.

Coloured Images

Please follow the link to download the
Coloured Images of the book:

https://rebrand.ly/5n5dmzg

We have code bundles from our rich catalogue of books and videos available at **https://github.com/bpbpublications**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Did you know that BPB offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.bpbonline.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at :

business@bpbonline.com for more details.

At **www.bpbonline.com**, you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on BPB books and eBooks.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Table of Contents

1. Getting Started with Excel 1

Introduction 1

Structure 1

Objectives 1

Using Excel 2

Conclusion 4

2. The 10 Most Popular Functions in Excel 5

Introduction 5

Structure 5

Objectives 6

SUM Function 6

Uses and applications 6

The Formula 6

Errors to avoid 7

IF function 8

Uses and applications 8

The formula 9

Errors to avoid 9

LOOKUP function 10

Uses and applications 10

The formula 10

Errors to avoid 10

VLOOKUP function 11

Uses and applications 11

The formula 12

Errors to avoid 12

MATCH function 13

Uses and applications *13*
The formula *13*
Errors to avoid *13*
CHOOSE function 14
Uses and applications *14*
The formula *14*
Errors to avoid *14*
DATE function 14
Uses and applications *15*
The formula *15*
Errors to avoid *15*
DAYS function 15
Uses and applications *15*
The formula *16*
Errors to avoid *16*
FIND and FINDB functions 16
Uses and applications *16*
The formula *16*
Errors to avoid *16*
INDEX function 17
Uses and applications *17*
The formula *17*
Errors to avoid *17*
Conclusion 18
Points to remember 18
Multiple choice questions 19
Answers *20*

3. Logical Functions in Excel 21
Introduction 21
Structure 21

Objectives 22
AND function 22
Uses and applications 22
The formula 22
Errors to avoid 23
FALSE function 23
Uses and applications 23
The formula 23
Errors to avoid 24
TRUE function 24
Uses and applications 24
The formula 24
Errors to avoid 25
IFERROR function 25
Uses and applications 25
The formula 25
Errors to avoid 26
IFNA function 26
Uses and applications 26
The formula 26
Errors to avoid 27
IFS function 27
Uses and applications 27
The formula 27
Errors to avoid 28
NOT function 28
Uses and applications 28
The formula 28
Errors to avoid 29
OR function 29
Uses and applications 29

The formula *29*
Errors to avoid *29*
XOR function 30
Uses and applications *30*
The formula *30*
Errors to avoid *30*
SWITCH function 30
Uses and applications *30*
The formula *30*
Errors to avoid *31*
Conclusion 31
Points to remember 31
Multiple choice questions 32
Answers *32*

4. Lookup and Reference Functions in Excel 33
Introduction 33
Structure 33
Objectives 34
ADDRESS function 34
Uses and applications *34*
The formula *34*
Errors to avoid *35*
AREAS function 35
Uses and applications *35*
The formula *35*
Errors to avoid *36*
COLUMN and COLUMNS functions 36
Uses and applications *36*
The formula *36*
Errors to avoid *37*

FILTER function 37
Uses and applications *37*
The formula *37*
Errors to avoid *37*
FORMULATEXT function 38
Uses and applications *38*
The formula *38*
Errors to avoid *38*
GETPIVOTDATA function 38
Uses and applications *38*
The formula *39*
Errors to avoid *39*
HLOOKUP function 39
Uses and applications *39*
The formula *39*
Errors to avoid *40*
HYPERLINK function 40
Uses and applications *40*
The Formula *40*
Errors to avoid *41*
INDIRECT function 41
Uses and applications *41*
The formula *41*
Errors to avoid *42*
OFFSET function 42
Uses and applications *43*
The formula *43*
Errors to avoid *43*
ROW and ROWS functions 43
Uses and applications *43*

The formula *44*
Errors to avoid *44*
RTD function 44
Uses and applications *44*
The formula *45*
Errors to avoid *45*
SORT and SORTBY functions 45
Uses and applications *45*
The formula *45*
Errors to avoid *46*
TRANSPOSE function 46
Uses and applications *46*
The formula *47*
Errors to avoid *47*
UNIQUE function 47
Uses and applications *47*
The formula *48*
Errors to avoid *48*
XLOOKUP function 48
Uses and applications *48*
The formula *48*
Errors to avoid *49*
XMATCH function 49
Uses and applications *49*
The formula *49*
Errors to avoid *49*
Conclusion 50
Points to remember 50
Multiple choice questions 51
Answers *51*

5. Math Functions in Excel 53
Introduction 53
Structure 53
Objectives 54
ABS function 55
Uses and applications 55
The formula 55
Errors to avoid 55
ACOS and ACOSH functions 55
Uses and applications 55
The formula 56
Errors to avoid 56
ACOT and ACOTH functions 56
Uses and applications 56
The formula 57
Errors to avoid 57
ASIN and ASINH functions 57
Uses and applications 57
The formula 58
Errors to avoid 58
ATAN and ATANH functions 58
Uses and applications 58
The formula 58
Errors to avoid 59
CEILING and FLOOR functions 59
Uses and applications 59
The formula 59
Errors to avoid 60
COMBIN and COMBINA functions 60
Uses and applications 60
The formula 60

Errors to avoid *61*
COS and COSH functions 61
Uses and applications *61*
The formula *61*
Errors to avoid *62*
COT and COTH functions 62
Uses and applications *62*
The formula *62*
Errors to avoid *63*
CSC and CSCH functions 63
Uses and applications *63*
The formula *63*
Errors to avoid *64*
DECIMAL function 64
Uses and applications *64*
The formula *64*
Errors to avoid *65*
DEGREES and RADIANS functions 65
Uses and applications *65*
The formula *65*
Errors to avoid *66*
EVEN and ODD functions 66
Uses and applications *66*
The formula *66*
Errors to avoid *66*
INT function 67
Uses and applications *67*
The formula *67*
Errors to avoid *67*
LCM function 67
Uses and applications *68*

The formula ... *68*
Errors to avoid ... *68*
LN and LOG functions ... 68
Uses and applications ... *68*
The formula ... *68*
Errors to avoid ... *69*
MMULT functions ... 69
Uses and applications ... *69*
The formula ... *69*
Errors to avoid ... *70*
MOD function ... 70
Uses and applications ... *70*
The formula ... *70*
Errors to avoid ... *70*
POWER function ... 71
Uses and applications ... *71*
The formula ... *71*
Errors to avoid ... *71*
PRODUCT function ... 71
Uses and applications ... *72*
The formula ... *72*
Errors to avoid ... *72*
RAND and RANDBETWEEN functions ... 72
Uses and applications ... *72*
The formula ... *73*
Errors to avoid ... *73*
ROMAN function ... 73
Uses and applications ... *74*
The formula ... *74*
Errors to avoid ... *74*
ROUND, ROUNDUP, and ROUNDDOWN functions ... 75

Uses and applications *75*
The formula *75*
Errors to avoid *76*
SEC and SECH functions 76
Uses and applications *76*
The formula *76*
Errors to avoid *76*
SIGN function 77
Uses and applications *77*
The formula *77*
Errors to avoid *77*
SIN and SINH function 77
Uses and applications *77*
The formula *78*
Errors to avoid *78*
SQRT function 78
Uses and applications *78*
The formula *78*
Errors to avoid *79*
SUBTOTAL function 79
Uses and applications *79*
The formula *79*
Errors to avoid *80*
SUMIF, SUMPRODUCT, and SUMSQ functions 80
Uses and applications *80*
The formula *81*
Errors to avoid *81*
TAN and TANH functions 81
Uses and applications *81*
The formula *82*
Errors to avoid *82*

Conclusion 82
Points to remember 82
Multiple choice questions 83
Answers *83*

6. Statistical Functions in Excel 85
Introduction 85
Structure 85
Objectives 86
NORM.DIST and NORM.INV functions 86
Uses and applications *87*
The formula *87*
Errors to avoid *87*
PERCENTILE.INC and PERCENTILE.EXC functions 88
Uses and applications *88*
The formula *88*
Errors to avoid *89*
PHI function 89
Uses and applications *89*
The formula *89*
Errors to avoid *89*
POISSON.DIST function 90
Uses and applications *90*
The formula *90*
Errors to avoid *90*
PROB function 90
Uses and applications *91*
The formula *91*
Errors to avoid *91*
QUARTILE.EXC and QUARTILE.INC functions 92
Uses and applications *92*

The formula ... *92*
Errors to avoid ... *93*
RANK.AVG and RANK.EQ functions ... 93
Uses and applications ... *93*
The formula ... *93*
Errors to avoid ... *94*
RSQ function ... 94
Uses and applications ... *94*
The formula ... *94*
Errors to avoid ... *94*
SKEW and SKEW.S functions ... 95
Uses and applications ... *95*
The formula ... *95*
Errors to avoid ... *96*
STDEVA and STDEVPA functions ... 96
Uses and applications ... *96*
The formula ... *96*
Errors to avoid ... *97*
FORECAST and FORECAST.LINEAR functions ... 97
Uses and applications ... *97*
The formula ... *97*
Errors to avoid ... *98*
GEOMEAN and HARMEAN functions ... 98
Uses and applications ... *98*
The formula ... 99
Errors to avoid ... *99*
GROWTH function ... 99
Uses and applications ... *99*
The formula ... *99*
Errors to avoid ... *100*
LOGNORM.DIST and LOGNORM.INV functions ... 100

Uses and applications *100*
The formula *100*
Errors to avoid *101*
MEDIAN function 101
Uses and applications *101*
The formula *101*
Errors to avoid *102*
MIN function 102
Uses and applications *102*
The formula *102*
Errors to avoid *103*
NEGBINOM.DIST function 103
Uses and applications *103*
The formula *103*
Errors to avoid *104*
STEYX function 104
Uses and applications *104*
The formula *104*
Errors to avoid *105*
T.DIST, T.DIST.2T, and T.DIST.RT functions 105
Uses and applications *105*
The formula *105*
Errors to avoid *106*
T.INV and T.INV.2T functions 106
Uses and applications *106*
The formula *106*
Errors to avoid *107*
T.TEST function 107
Uses and applications *107*
The formula *107*
Errors to avoid *108*

TREND function 108
Uses and applications *108*
The Formula *108*
Errors to avoid *109*
TRIMMEAN function 109
Uses and applications *109*
The formula *109*
Errors to avoid *110*
VAR.P and VAR.S functions 110
Uses and applications *110*
The formula *110*
Errors to avoid *111*
Z.TEST function 111
Uses and applications *111*
The formula *111*
Errors to avoid *112*
Conclusion 112
Points to remember 112
Multiple choice questions 113
Answers *113*

7. Text Functions in Excel 115
Introduction 115
Structure 115
Objectives 116
CLEAN function 116
Uses and applications *116*
The formula *117*
Errors to avoid *117*
CODE function 117
Uses and applications *117*

The formula ... *117*
Errors to avoid ... *118*
CONCATENATE function ... 118
Uses and applications ... *118*
The formula ... *118*
Errors to avoid ... *118*
DOLLAR function ... 118
Uses and applications ... *119*
The formula ... *119*
Errors to avoid ... *119*
EXACT function ... 119
Uses and applications ... *119*
The formula ... *119*
Errors to avoid ... *120*
FIND function ... 120
Uses and applications ... *120*
The formula ... *120*
Errors to avoid ... *121*
FIXED function ... 121
Uses and applications ... *121*
The formula ... *121*
Errors to avoid ... *121*
LEFT and RIGHT functions ... 121
Uses and applications ... *122*
The formula ... *122*
Errors to avoid ... *122*
LEN function ... 122
Uses and applications ... *123*
The formula ... *123*
Errors to avoid ... *123*
LOWER, UPPER functions ... 123

Uses and applications *123*
The formula *123*
Errors to avoid *124*
MID function 124
Uses and applications *124*
The formula *124*
Errors to avoid *125*
NUMBERVALUE function 125
Uses and applications *125*
The formula *125*
Errors to avoid *125*
PROPER function 125
Uses and applications *126*
The formula *126*
Errors to avoid *126*
REPLACE function 126
Uses and applications *126*
The formula *126*
Errors to avoid *127*
SEARCH function 127
Uses and applications *127*
The formula *127*
Errors to avoid *128*
SUBSTITUTE function 128
Uses and applications *128*
The Formula *128*
Errors to avoid 128
T function *128*
Uses and applications *129*
The formula *129*
Errors to avoid *129*

TEXT function 129
Uses and applications *129*
The formula *129*
Errors to avoid *130*
TRIM function 130
Uses and applications *130*
The formula *130*
Errors to avoid *130*
UNICHAR function 130
Uses and applications *131*
The formula *131*
Errors to avoid *131*
UNICODE function 131
Uses and applications *131*
The formula *131*
Errors to avoid *132*
VALUE function 132
Uses and applications *132*
The formula *132*
Errors to avoid *132*
Conclusion 133
Points to remember 133
Multiple choice questions 133
Answers *134*

8. Information Functions in Excel 135
Introduction 135
Structure 135
Objectives 136
Cell function 136
Uses and applications *136*

The formula *136*
Errors to avoid *137*
ERROR.TYPE function 137
Uses and applications *137*
The formula *137*
Errors to avoid *138*
INFO function 138
Uses and applications *138*
The formula *138*
Errors to avoid *139*
ISBLANK function 139
Uses and applications *139*
The formula *139*
Errors to avoid *139*
ISERROR and ISERR functions 140
Uses and applications *140*
The formula *140*
Errors to avoid *140*
ISEVEN and ISODD functions 140
Uses and applications *141*
The formula *141*
Errors to avoid *141*
ISFORMULA function 141
Uses and applications *141*
The formula *141*
Errors to avoid *142*
ISLOGICAL function 142
Uses and applications *142*
The formula *142*
Errors to avoid *142*
ISNA function 143

Uses and applications *143*
The formula *143*
Errors to avoid *143*
ISNUMBER function 143
Uses and applications *143*
The formula *144*
Errors to avoid *144*
ISREF function 144
Uses and applications *144*
The formula *144*
Errors to avoid *145*
ISTEXT function 145
Uses and applications *145*
The formula *145*
Errors to avoid *145*
N function 145
Uses and applications *146*
The formula *146*
Errors to avoid *146*
SHEET function 146
Uses and applications *146*
The formula *147*
Errors to avoid *147*
Type function 147
Uses and applications *147*
The formula *147*
Errors to avoid *148*
Conclusion 148
Points to Remember 148
Multiple choice questions 149
Answers *149*

9. Financial Functions in Excel 151
Introduction 151
Structure 151
Objectives 152
COUPDAYS, COUPNUM, and COUPPCD functions 152
Uses and applications 152
The formula 153
Errors to avoid 153
CUMIPMT and CUMPRINC functions 153
Uses and applications 154
The formula 154
Errors to avoid 154
DB and DDB functions 155
Uses and applications 155
The formula 155
Errors to avoid 155
DISC function 156
Uses and applications 156
The formula 156
Errors to avoid 157
DURATION and MDURATION functions 157
Uses and applications 157
The formula 157
Errors to avoid 158
EFFECT function 158
Uses and applications 158
The formula 158
Errors to avoid 158
Time Value of money functions 158
Uses and applications 158
The formula 159

Errors to avoid *160*
INTRATE function 160
Uses and applications *160*
The formula *160*
Errors to avoid *161*
IPMT function 161
Uses and applications *161*
The formula *161*
Errors to avoid *162*
IRR, MIRR, and NPV functions 162
Uses and applications *162*
The formula *162*
Errors to avoid *163*
NOMINAL function 163
Uses and applications *163*
The formula *164*
Errors to avoid *164*
PRICE, YIELD, YIELDDISC, and YIELDMAT functions 164
Uses and applications *164*
The formula *164*
Errors to avoid *165*
XIRR and XNPV functions 166
Uses and applications *166*
The formula *166*
Errors to avoid *167*
VDB function 167
Uses and applications *167*
The formula *167*
Errors to avoid *168*
Conclusion 168
Multiple choice questions 168

Answers *169*

10. Date and Time Functions in Excel 171

Introduction 171

Structure 171

Objectives 172

DATE function 172

Uses and applications *172*

The formula *172*

Errors to avoid *173*

DAY, MONTH, and YEAR functions 173

Uses and applications *173*

The formula 173

Errors to avoid *173*

DAYS and DAYS360 functions 174

Uses and applications *174*

The formula *174*

Errors to avoid *174*

EDATE and EOMONTH function 174

Uses and applications *175*

The formula *175*

Errors to avoid *175*

HOUR, MINUTE, and SECOND functions 175

Uses and applications *175*

The formula *176*

Errors to avoid *176*

ISOWEEKNUM and WEEKNUM functions 176

Uses and applications *177*

The formula *177*

Errors to avoid *177*

NETWORKDAYS function 177

Uses and applications *177*
The formula *178*
Errors to avoid *178*
NOW and TODAY functions 178
Uses and applications *178*
The formula *178*
Errors to avoid *179*
TIME function 179
Uses and applications *179*
The formula *179*
Errors to avoid *179*
WEEKDAY function 180
Uses and applications *180*
The formula *180*
Errors to avoid *180*
Conclusion 180
Points to remember 181
Multiple choice questions 181
Answers 182

11. Database Functions in Excel 183
Introduction 183
Structure 183
Objectives 184
DAVERAGE function 184
Uses and applications *184*
The formula *184*
Errors to avoid *185*
DCOUNT function 185
Uses and applications *185*
The formula *185*

Errors to avoid *186*
DCOUNTA function 186
Uses and applications *186*
The formula *186*
Errors to avoid *187*
DGET function 187
Uses and applications *187*
The formula *187*
Errors to avoid *188*
DMIN and DMAX functions 188
Uses and applications *188*
The formula *188*
Errors to avoid *189*
DPRODUCT function 189
Uses and applications *189*
The formula *189*
Errors to avoid *190*
DSUM function 190
Uses and applications *190*
The formula *190*
Errors to avoid *191*
DSTDEV and DVAR functions 191
Uses and applications 191
The Formula 192
Errors to avoid *192*
Conclusion 192
Multiple choice questions 193
Answers *193*

12. Cube Functions in Excel 195
Introduction 195

Structure 195
Objectives 196
CUBEMEMBER function 196
Uses and applications *196*
The formula *196*
Errors to avoid *197*
CUBEKPIMEMBER function 197
Uses and applications *197*
The formula *197*
Errors to avoid *197*
CUBEMEMBERPROPERTY function 197
Uses and applications *198*
The formula *198*
Errors to avoid *198*
CUBERANKEDMEMBER function 198
Uses and applications *198*
The formula *198*
Errors to avoid *199*
CUBEVALUE function 199
Uses and applications *199*
The formula *199*
Errors to avoid *199*
Conclusion 199
Multiple choice questions 200
Answers *200*

13. Web Functions in Excel 201
Introduction 201
Structure 201
Objectives 201
ENCODEURL function 202

Uses and applications *202*
The formula *202*
Errors to avoid *202*
FILTERXML function 203
Uses and applications *203*
The formula *203*
Errors to avoid *203*
WEBSERVICE function 204
Uses and applications *204*
The formula *204*
Errors to avoid *204*
Conclusion 204
Multiple choice questions 204
Answers *205*

14. User-defined Functions in Excel 207
Introduction 207
Structure 207
Objectives 207
CALL function 208
Uses and applications *208*
The formula *208*
Errors to avoid *208*
EUROCONVERT function 209
Uses and applications *209*
The formula *210*
Errors to avoid *211*
REGISTER.ID function 211
Uses and applications *212*
The formula *212*
Errors to avoid *212*

Conclusion 212
Multiple choice questions 213
Answers *213*

Index 215-231

Chapter 1
Getting Started with Excel

Introduction

Welcome to this Excel journey. This book is designed as a reference to help business professionals and students who wish to use Excel to advance their career opportunities and academic studies. The goal of the book is to help you understand over 200 different Excel formulas that can be used for many different applications. In this chapter, we will talk about the structure of the book, how to get the most out of it, and why Excel is valuable for your career.

Structure

In this chapter, we will discuss the following topics:

- Using Excel

Objectives

In this book, we will refer to Excel functions and formulas—the terms may seem interchangeable, but there is a slight difference between them. The function is the larger tool; for instance, the SUM function allows one to sum up a set of numbers. The formula is the specific language that enables the function. The point is largely

academic, but it is included here in the hopes of avoiding any confusion when you are reading and seeing these two terms used.

As we examine each function, we will spell out what the formula is behind the function, what the function is used for, the specific data that is required in Excel to make the function work properly, an example of the function in practice, and finally, images from Excel illustrating the function being used.

It is important to note that different versions of Excel look slightly different, and in a few cases, older versions of Excel may not have some new functions created for more recent versions of Excel. We have tried to pick functions that are the most useful for all users, but some inconsistency between different versions of Excel is unavoidable.

Using Excel

Before we begin examining the functions and formulas of Excel, it is worth taking a moment and reviewing the basics of Excel.

Excel is a spreadsheet program with some database features. It is the most widely used business software tool in the world (assuming one excludes operating systems like Windows from that definition, of course). The reason Excel is so popular is that it is versatile yet simple to use. There are many pieces of software that are more powerful than Excel and many that offer tools and features that Excel does not have. Yet Excel's popularity persists because it is intuitive and powerful enough for most basic business tasks—and this includes everything from basic accounting to mathematical calculations to building small datasets.

Excel starts the user with a spreadsheet with a series of rows and columns. These rows and columns can be used to enter data—either quantitative data like numbers and dates or qualitative data such as names, places, notes, or any sort.

Excel starts with a single page or "sheet" of data, but the user can easily add additional workbook sheets with the **+** that is displayed to the right of the sheet name. In *figure 1.1*, we can see that there are two workbooks:

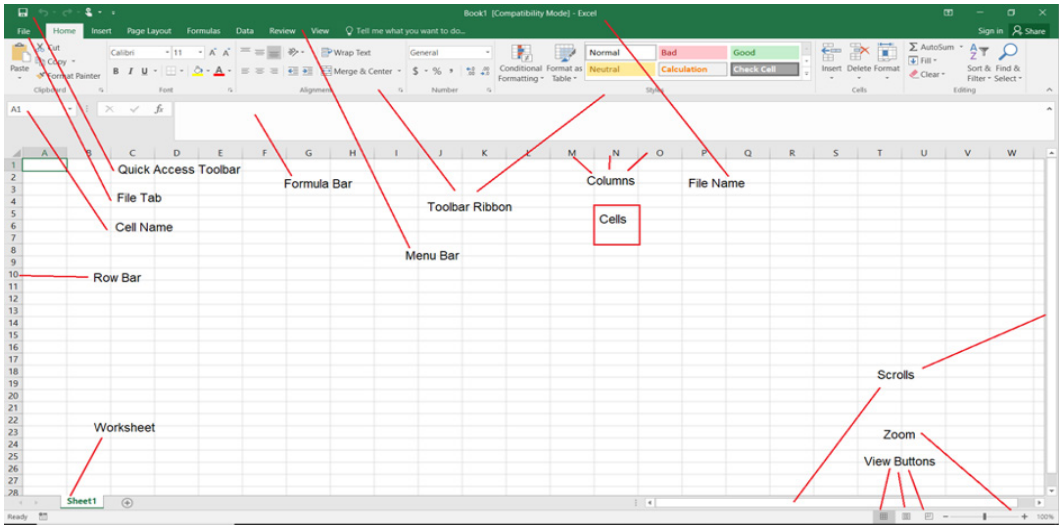


Figure 1.1: *The basic Excel Workbook*

The formulas in Excel are entered into a particular chosen square or "cell," and that cell formula then refers to or "references" other cells either in that sheet or in another sheet of data within the workbook.

TIP: In general, where possible, it is a good idea to keep formulas and the cells they reference on the same sheet to make the Excel model more readable.

Excel's versatility allows you to build many different types of documents, from inventory lists to small datasets to financial statements. When you finish an Excel document or "model," as they are often called, it could look something like *figure 1.2*:

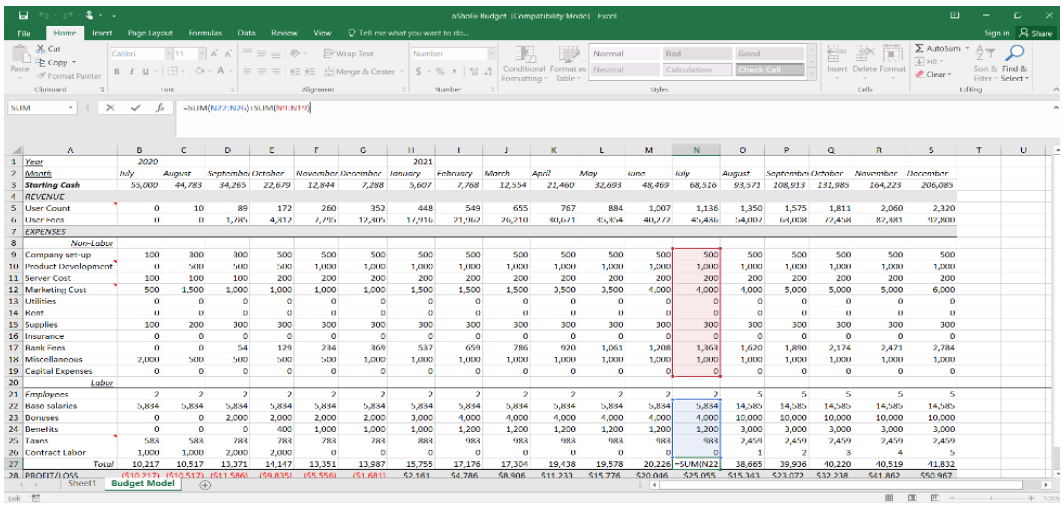


Figure 1.2: *An example Excel model*

In this example, we have built a budget for a company—as we see here, the budget includes time periods in the form of months in each column and specific budget line items in each row. The cells are a combination of numbers inputted into the sheet and formulas derived from other cells, as shown preceding. In Excel, when you click on a formula, you are automatically shown other cells that the formula references.

The end product may look daunting, but once you get started with Excel, you will quickly find that it is not as difficult as it seems. Ready to give it a try on your own? Then let us jump in!

Conclusion

In this chapter, we got an introduction to Excel. We looked at the basic Excel workbook as well as an example Excel model.

In the upcoming chapters, we will dive deeper into functions in Excel that will surely prove to be very helpful to you.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 2

The 10 Most Popular Functions in Excel

Introduction

You do not need to memorize hundreds of Excel formulas to start being quite proficient with the software. The truth is that for any given job, you will likely use the same 10–20 formulas repeatedly as you perform that job. In this chapter, we will get you started with some of the most common formulas that you may need to know based on the 10 most popular and widely used formulas in Excel.

Structure

In this chapter, we will discuss the following topics:

- SUM function
- IF function
- LOOKUP function
- VLOOKUP function
- MATCH function
- CHOOSE function
- DATE function

- DAYS function
- FIND, FINDB functions
- INDEX function

Objectives

After studying this unit, you will be able to use the most common Excel functions in Excel and determine the appropriate situations to apply these formulas.

SUM Function

The **SUM** function is one of the most popular Excel functions. It basically allows the user to quickly and easily add up or "sum" a series of numbers together. This is very useful if you have rows or columns of figures you want to add and then display a total of that row or column in another part of the spreadsheet.

Uses and applications

Use the SUM function to add up a series of numbers, for example, sales made by different locations or on different days. **SUM** can be used for either rows or columns of figures. You will find it used frequently in financial spreadsheets as well as in any spreadsheet tracking data over time. **SUM** combines well with the **AVERAGE**, **MIN**, and **MAX** functions when you are trying to summarize a set of data, and it makes it easy for the user to understand what is happening in that data without having to review every cell individually.

The Formula

=SUM(Reference Cells)

Figure 2.1 illustrates the use of the **SUM** function:

2/4/2022	3,101	4,028	7,569	3,487	4,676
2/11/2022	5,360	7,009	11,193	6,741	6,587
2/18/2022	4,472	6,097	10,544	5,437	6,259
2/25/2022	3,720	5,820	10,289	5,196	5,578
3/4/2022	5,210	6,673	11,808	5,018	6,286
3/11/2022	4,646	7,282	9,776	5,280	6,427
3/18/2022	4,494	6,310	10,863	5,373	6,995
3/25/2022	4,606	7,233	10,068	5,860	6,290
4/1/2022	4,621	6,709	10,370	5,403	5,999
4/8/2022	5,172	7,658	10,691	6,135	7,296
4/15/2022	5,813	7,762	11,602	6,398	6,771
4/22/2022	4,394	6,625	10,333	4,718	5,195
SUM	=SUM(D165:D176)				

Figure 2.1: *The sum function*

Errors to avoid

Oftentimes **SUM** will be used in one row or column and then dragged over or down to be applied to other rows or columns. "Dragging and dropping" in Excel refers to applying a formula like **SUM** from one cell to another cell. To do this, highlight the cell or cells you want to drag and drop, then click hold down the bottom right corner of the selection box (you will see a tiny square in the corner, and your cursor shape will change to a cross), and finally, drag your mouse along the cells that you want the formula applied to while holding down the left mouse button. When you have finished dragging, release the mouse button, and the cells will update with the formula. The process seems confusing when described in words, but after you have tried it a couple of times, you will find it is actually very easy. However, you need to make sure that the new cells the **SUM** function is being applied to contain the correct data you are interested in.

If some data from other columns is missing or is a different data type, as in *figure 2.2*, it can cause problems in your spreadsheet:

1/7/2022	2,288	2,626		4,234	2,419
1/14/2022	2,946	2,755		5,532	3,763
1/21/2022	3,121	3,089		6,011	3,451
1/28/2022	3,603	3,844		6,340	4,127
2/4/2022	2,910	3,341		5,097	3,074
2/11/2022	4,380	4,440		7,494	4,727
2/18/2022	4,110	3,234		7,951	4,266
2/25/2022	3,305	3,745		5,610	3,510
3/4/2022	4,986	3,981		7,087	3,990
3/11/2022	3,787	3,895		7,004	3,816
3/18/2022	3,737	3,901		7,349	4,541
3/25/2022	4,412	4,968	2,112	6,996	4,175
4/1/2022	3,697	4,359	2,229	6,971	4,328
4/8/2022	4,642	5,693	2,540	7,419	5,359
4/15/2022	4,727	5,831	2,936	8,145	5,331
4/22/2022	3,836	4,173	2,193	5,829	3,888
SUM	52,132	(64:E176)	12,010	89,292	55,132

Figure 2.2: *The sum function in practice*

Similarly, if you are going to anchor your cells, be sure that any dragging of the cells applies SUM to the correct reference area afterwards. Refer to the following figure:

SUM =SUM(D3:H3)

	A	D	E	F	G	H	I
1	Store	D	E	F	G	H	Total
2	Sum	973568	724940	661492	705116	838805	
3	Average	8,813.21	5,494.08	4,291.78	5,135.95	5,977.35	=SUM(D3:H3)
4	Rank	1	3	5	4	2	
5	1/11/2019	4,644	4,035	3,533	4,170	5,158	21,540
6	1/18/2019	5,084	4,042	3,784	4,038	5,762	22,710
7	1/25/2019	5,243	4,162	3,725	4,525	6,166	23,821
8	2/1/2019	4,320	3,298	3,007	3,942	4,658	19,225
9	2/8/2019	5,299	4,207	3,575	4,103	5,178	22,362
10	2/15/2019	6,659	4,846	4,372	4,662	6,253	26,792
11	2/22/2019	5,346	4,223	3,690	4,052	5,456	22,767
12	3/1/2019	5,161	4,649	4,144	4,438	5,248	23,640
13	3/8/2019	5,381	4,376	3,991	3,736	5,863	23,347
14	3/15/2019	5,714	4,259	4,006	4,348	5,241	23,568
15	3/22/2019	5,089	4,621	4,788	4,633	6,117	25,248
16	3/29/2019	5,650	5,066	4,531	4,384	6,577	26,208
17	4/5/2019	5,549	5,005	3,979	4,359	6,301	25,193
18	4/12/2019	6,196	4,389	4,986	4,724	5,452	25,747
19	4/19/2019	5,647	4,906	4,744	4,751	5,674	25,722

Figure 2.3: *The sum function used horizontally*

IF function

The IF function is one of the most useful functions in all of Excel. Essentially, it lets the user go through and create a simple IF-THEN type statement based on other cells in an Excel sheet. This lets us flag cells that are of particular interest from a large mass of data. For example, we might use the **`IF`** function to find the first data point in a set of panel data. But we can also take this one step further and combine **`IF`** statements together to create a "nested" IF statement.

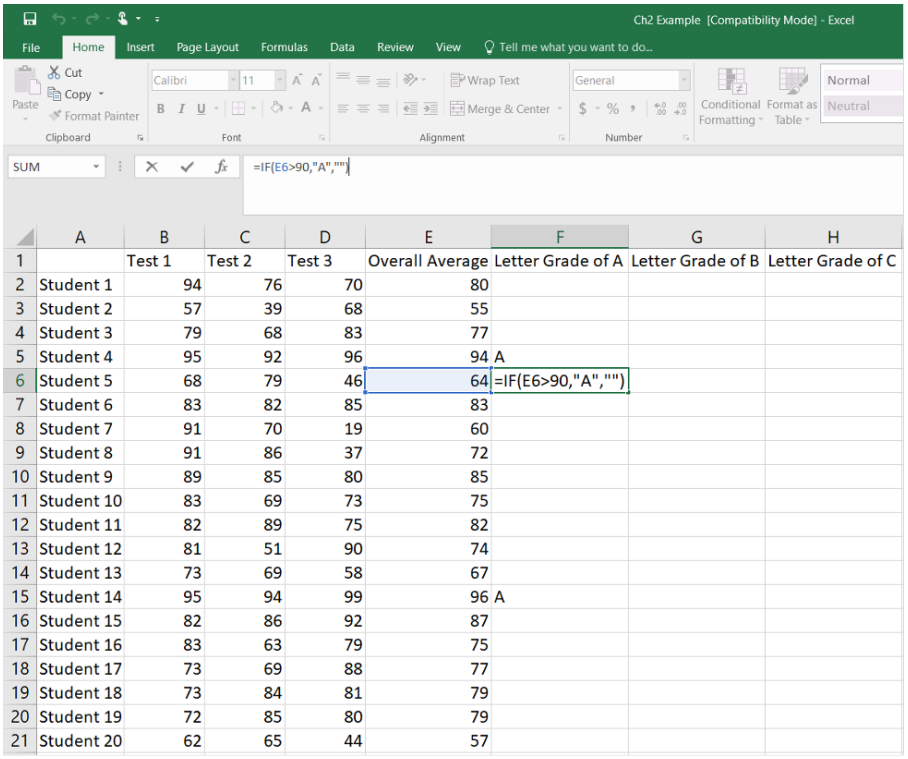
Uses and applications

Functions like **`SUM`** are very useful for doing a specific but common task, like adding up a set of numbers. However, they are not very versatile. In contrast, the **`IF`** function can be used for many different tasks. For instance, we might use the **`IF`** function to assign letter grades to a student in a class based on test scores, identify sales accounts in a business that need more scrutiny based on their purchase volume, or find physical addresses that meet a certain condition in a set of contacts. **`IF`** can be used with many logical or arithmetic functions in hundreds of handy ways.

The formula

=IF(Reference Cells and condition, Value if condition is true, Value if condition is false)

The following figure shows the IF function being applied in Excel:



	A	B	C	D	E	F	G	H
1		Test 1	Test 2	Test 3	Overall Average	Letter Grade of A	Letter Grade of B	Letter Grade of C
2	Student 1	94	76	70	80			
3	Student 2	57	39	68	55			
4	Student 3	79	68	83	77			
5	Student 4	95	92	96	94	A		
6	Student 5	68	79	46	64	=IF(E6>90,"A",""")		
7	Student 6	83	82	85	83			
8	Student 7	91	70	19	60			
9	Student 8	91	86	37	72			
10	Student 9	89	85	80	85			
11	Student 10	83	69	73	75			
12	Student 11	82	89	75	82			
13	Student 12	81	51	90	74			
14	Student 13	73	69	58	67			
15	Student 14	95	94	99	96	A		
16	Student 15	82	86	92	87			
17	Student 16	83	63	79	75			
18	Student 17	73	69	88	77			
19	Student 18	73	84	81	79			
20	Student 19	72	85	80	79			
21	Student 20	62	65	44	57			

Figure 2.4: *The IF function*

Errors to avoid

While the preceding example is a fairly simple one, it is easy to make mistakes with IF statements, especially when they are used as part of a nested IF block or used with other functions. Be sure to check the results after you apply an IF statement to make sure it is giving you what you expect. For instance, in the preceding example, we are using an IF statement to determine those people with a test average above 90 and then give them an A grade. For anyone below 90, no letter is included in the cell based on the “” that appears at the end of the formula. That might be an issue, of course, if we want to report all the grades in one column.

When using more advanced IF functions, we need to be careful about our conditions, as the formulas can become hard to read. This may be unavoidable, though, if we are trying to perform a relatively complex computation using Excel, shown as follows.

Here in *figure 2.5*, we use a nested IF statement to assign a letter grade based on a series of conditions:

SUM =IF(E9>90,"A",IF(E9>80,"B",IF(E9>70,"C", "F")))

	A	B	C	D	E	F
1		Test 1	Test 2	Test 3	erall Avera	Class Grade
2	Student 1	94	76	70	80	B
3	Student 2	57	39	68	55	F
4	Student 3	79	68	83	77	C
5	Student 4	95	92	96	94	A
6	Student 5	68	79	46	64	F
7	Student 6	83	82	85	83	B
8	Student 7	91	70	19	60	F
9	Student 8	91	86	37	72	=IF(E9>90,"A",IF(E9>80,"B",IF(E9>70,"C", "F")))
10	Student 9	89	85	80	85	B
11	Student 10	83	69	73	75	C
12	Student 11	82	89	75	82	B
13	Student 12	81	51	90	74	C
14	Student 13	73	69	58	67	F
15	Student 14	95	94	99	96	A
16	Student 15	82	86	92	87	B
17	Student 16	83	63	79	75	C
18	Student 17	73	69	88	77	C
19	Student 18	73	84	81	79	C
20	Student 19	72	85	80	79	C
21	Student 20	62	65	44	57	F

Figure 2.5: *A more complex IF function*

LOOKUP function

The LOOKUP function is similar to the **SUM** function in that it does a simple but very common task—it looks up a value in a sheet based on a reference.

Uses and applications

LOOKUP functions are generally used when dealing with large datasets and the need to find something in the data without having to search for it manually or perform a *Ctrl + F*. They are very useful for displaying a dynamic value that changes as we change, say, a part number in inventory or a store number from a list of locations.

The formula

=LOOKUP(Value to be looked up, Array of data to search for the value)

Errors to avoid

The most common error that comes up when dealing with lookup values is that of failing to put the array in properly. **LOOKUP** is somewhat less useful than the newer

function **VLOOKUP**. The reason for this is that **LOOKUP** has a hard time returning the correct value that the user is actually looking for in a dataset. Instead, if faced with a series of columns, **LOOKUP** will return the value from the far right column in the array regardless of which column the user is actually focused on. This makes it important to be careful picking the array.

Similarly, if there are multiple instances of the value to be looked up, the **LOOKUP** function does not give the user any indication of the problem, so it is important to be very careful that your lookup value is a unique identifier and does not have duplicates in the data. Refer to the following figure to see the use of the **LOOKUP** function:

SUM =LOOKUP(H2,A2:E27)

	A	B	C	D	E	F	G	H
1	Part ID:	Price	Cost Per Unit	Weight	Inventory on Hand			
2	A	61.00	4.27	27.00	16.00		PART	C
3	B	5.00	4.16	24.00	15.00		PRICE	52
4	C	52.00	8.34	76.00	20.00		STOCK AVAILBLE	=LOOKUP(H2,A2:E27)
5	D	54.00	48.99	22.00	13.00			
6	E	91.00	12.74	52.00	15.00			
7	F	68.00	67.94	43.00	19.00			
8	G	40.00	28.70	4.00	18.00			
9	H	82.00	55.62	27.00	5.00			
10	I	59.00	58.81	64.00	10.00			
11	J	30.00	10.21	52.00	9.00			
12	K	48.00	26.94	42.00	12.00			
13	L	20.00	14.44	64.00	15.00			
14	M	97.00	85.00	43.00	8.00			
15	N	41.00	1.44	38.00	19.00			
16	O	96.00	79.70	75.00	20.00			
17	P	72.00	16.08	86.00	18.00			
18	Q	68.00	46.78	84.00	16.00			

Figure 2.6: *The Lookup function*

VLOOKUP function

The **VLOOKUP** function, along with the IF function, is probably one of the most versatile tools in Excel and also one of the most widely used in more advanced Excel models. In essence, **VLOOKUP** lets the user lookup values, but without the limitation of the standard LOOKUP function. The V in **VLOOKUP** stands for vertical because the function looks for a value in a specific vertical column, as opposed to **HLOOKUP** or horizontal lookup, with looks across a row of values.

Uses and applications

VLOOKUP is useful not only for looking up a single value like a product price but also for building a table that looks up values from a larger set of data—for instance, we might look up prices for half a dozen parts a customer is buying in order to determine the price of the customer's complete order.

The formula

=VLOOKUP(cell with value to be looked up, the set of data to be searched for the value, the column number for the data to be returned, approximate or exact match represented by a 1 or 0)

Errors to avoid

The two most common errors when using **VLOOKUP** are as follows:

- Not anchoring cells in the range of values to be searched can result in your search array sliding down as you drag the formula to be applied to a range of cells.
- Specifying an approximate match at the end of the formula rather than an exact match. An approximate match only matches the value to be looked up with the search array based on the first few characters of the lookup value—it is very rare that an approximate match should be used. Unless you have a good reason to use an approximate match, most Excel users are better off using the Exact match "0" option in the formula.

An example of the use of the **VLOOKUP** function is as follows:

SUM | =VLOOKUP(H7,$A$1:$E$27,2,0)

	A	B	C	D	E	F	G	H	I
1	Part ID:	Price	Cost Per Unit	Weight	Inventory on Hand		CUSTOMER ORDER		
2	A	61.00	4.27	27.00	16.00			Part ID	PRICE
3	B	5.00	4.16	24.00	15.00		PART 1	A	61
4	C	52.00	8.34	76.00	20.00		PART 2	C	52
5	D	54.00	48.99	22.00	13.00		PART 3	O	96
6	E	91.00	12.74	52.00	15.00		PART 4	G	40
7	F	68.00	67.94	43.00	19.00		PART 5	T	=VLOOKUP(H7,$A$1:$E$27,2,0)
8	G	40.00	28.70	4.00	18.00				
9	H	82.00	55.62	27.00	5.00		Total Cost		336
10	I	59.00	58.81	64.00	10.00				
11	J	30.00	10.21	52.00	9.00				
12	K	48.00	26.94	42.00	12.00				
13	L	20.00	14.44	64.00	15.00				
14	M	97.00	85.00	43.00	8.00				
15	N	41.00	1.44	38.00	19.00				
16	O	96.00	79.70	75.00	20.00				
17	P	72.00	16.08	86.00	18.00				
18	Q	68.00	46.78	84.00	16.00				
19	R	11.00	9.52	29.00	6.00				
20	S	14.00	3.10	27.00	6.00				
21	T	87.00	49.14	3.00	8.00				
22	U	49.00	7.29	100.00	16.00				
23	V	79.00	47.02	28.00	20.00				

Figure 2.7: *The VLOOKUP function*

MATCH function

The MATCH function is primarily used in conjunction with other functions to perform calculations or do analysis in Excel. Basically, **MATCH** allows the user to determine what position a specific value is in. For instance, one could use **MATCH** to find out what is the rank of an individual in a list. There is a newer version of **MATCH** called **XMATCH** that is available for those with newer versions of Excel.

Uses and applications

MATCH is most useful when dealing with large datasets in Excel and trying to extract information from that dataset in a dynamic way. For instance, if we had a list of all inventory products and we wanted to find where in the list a specific subset of products fall, we could use MATCH.

The formula

=MATCH(value to be looked up, set of data to be searched, less then, greater then, or exact match indicator)

Errors to avoid

As with **VLOOKUP**, there is a lot of confusion around the last term in the formula—the less than, greater than, and exact match indicators can often work in unexpected ways. Make sure you check the results to be sure you understand what the formula is doing.

An example of the use of the **MATCH** function is as follows:

SUM | =MATCH(H5,D1:D17,1)

	A	B	C	D	E	F	G	H
1	Part ID:	Price	Cost Per Unit	Weight	Inventory on Hand			
2	T	87.00	49.14	3.00	8.00		Item	B
3	G	40.00	28.70	4.00	18.00		Position	5
4	D	54.00	48.99	22.00	13.00			
5	B	5.00	4.16	24.00	15.00		Weight	10
6	A	61.00	4.27	27.00	16.00		Position	=MATCH(H5,D1:D17,1)
7	H	82.00	55.62	27.00	5.00			
8	S	14.00	3.10	27.00	6.00			
9	V	79.00	47.02	28.00	20.00			
10	R	11.00	9.52	29.00	6.00			
11	N	41.00	1.44	38.00	19.00			
12	K	48.00	26.94	42.00	12.00			
13	F	68.00	67.94	43.00	19.00			
14	M	97.00	85.00	43.00	8.00			
15	X	72.00	19.66	48.00	16.00			
16	E	91.00	12.74	52.00	15.00			
17	J	30.00	10.21	52.00	9.00			

Figure 2.8: *The match function*

CHOOSE function

The CHOOSE function is similar to a set of nested IF formulas—it enables the user to specify what should be done if a cell meets certain criteria.

Uses and applications

The **CHOOSE** function is often used with a toggle or dropdown in Excel to allow a financial model to change based on user inputs, such as assumptions about economic conditions or interest rates.

The formula

=CHOOSE(Reference cell where the input number is, value to be used when input is 1, value to be used when input value is 2, value to be used when input value is 3, and so on)

Errors to avoid

The most common error with **CHOOSE** is probably an error or mismatch that occurs when an input value if used—for instance, the user wants to show growth of 2% and instead shows growth of 3% simply because they had not been careful in setting up the formula.

An example of the use of **CHOOSE** function is as follows:

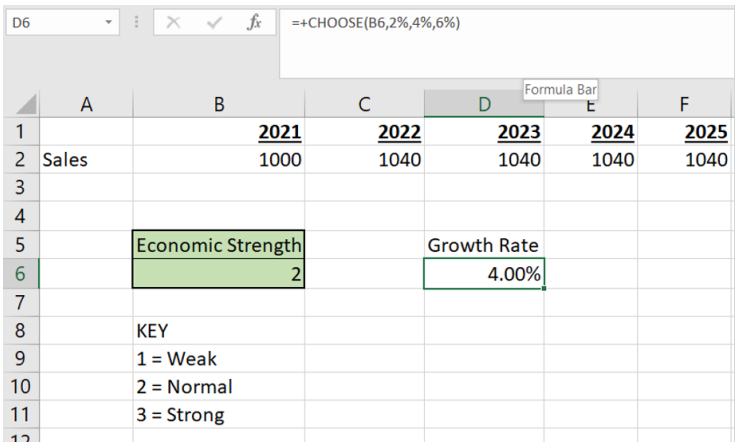


Figure 2.9: *The choose function*

DATE function

The DATE function allows you to create a date out of numbers representing days, months, and years.

Uses and applications

In a lot of data sets, the day, month, and year may be in separate columns. Excel's **DATE** function allows them to be combined. The real use of the function is that computers often have trouble translating dates between programs, so **DATE** allows us to avoid that by reducing the date to its components in one program and then reassembling them into a date in Excel. Similarly, using other Excel functions such as **LEFT**, **RIGHT**, or **CHOOSE**, we may generate values for days, months, and years and then need to put them into a valid date format using the **DATE** function.

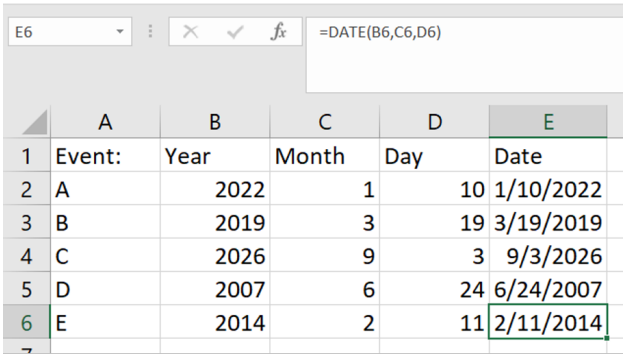
The formula

=DATE(Cell referring to Year, Cell Referring to Month, Cell Referring to Day)

Errors to avoid

DATE is a pretty simple but effective formula—just be careful to make sure you have your cells chosen correctly in the formula.

An example of the use of the **DATE** function is as follows:



	A	B	C	D	E
1	Event:	Year	Month	Day	Date
2	A	2022	1	10	1/10/2022
3	B	2019	3	19	3/19/2019
4	C	2026	9	3	9/3/2026
5	D	2007	6	24	6/24/2007
6	E	2014	2	11	2/11/2014

Figure 2.10: *The Date function*

DAYS function

Many Excel functions work together in combination with one another. The **DAYS** function and **DATE** function are examples of this. **DAYS** simply determine the number of days between two separate dates.

Uses and applications

DAYS is commonly used when we want to know the number of days between dates for a calculation, like average sales per day or interest accrued on loan.

The formula

=DAYS(ending date, starting date)

Errors to avoid

The **DAYS** formula is fairly simple, so if a mistake occurs, it is usually a matter of either having the date in the wrong format or mixing up the starting date and the ending date.

An example of the use of the **DAYS** function is as follows:

F6 =DAYS(E6,E5)

	A	B	C	D	E	F
1	Event:	Year	Month	Day	Date	Days Between
2	D	2007	6	24	6/24/2007	
3	E	2014	2	11	2/11/2014	2,424
4	B	2019	3	19	3/19/2019	1,862
5	A	2022	1	10	1/10/2022	1,028
6	C	2026	9	3	9/3/2026	1,697

Figure 2.11: *The Days function*

FIND and FINDB functions

FIND and **FINDB** are Excel functions to find the starting position of a set of characters in a text string. The **FIND** and **FINDB** functions are alternatives used with different languages—single-byte character sets and doubles by character sets, respectively.

Uses and applications

One common use of Excel is for dealing with semi-structured data—sets of data that are not in well-defined groups or fields. In this type of data, it is very common to have fields of text that are difficult to organize and use. **FIND** can help deal with this type of problem by starting to let us organize the data.

The formula

=FIND(Text to be found, Text to search, starting position in the text to search from). The last term is optional and set to 1, meaning the start of the text, by default.

Errors to avoid

If the text being searched for is not found, then Excel will return **#VALUE** in the cell. This happens frequently in large datasets, so be careful that you are looking for text

that does exist within the string being searched. Note also that **FIND** returns the first occurrence of the text being sought—so searching for the letter "a" in banana returns a value of 2 (second letter) rather than 4 or 6. **FIND** is also case-sensitive, so it will not return a value for "A" in bananas.

An example of the use of the **FIND** function is as follows:

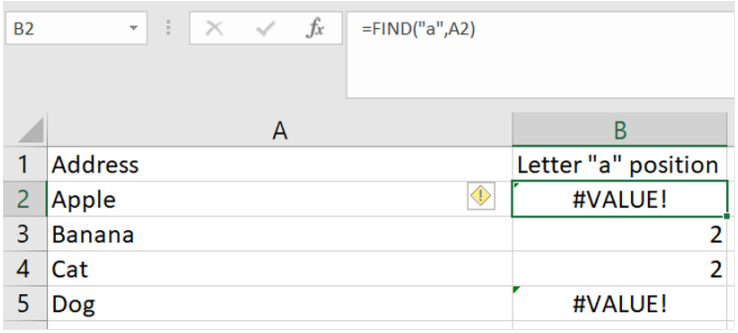


Figure 2.12: *The Find function*

INDEX function

The INDEX function is sort of like a GPS system for Excel—if you give Excel a set of "coordinates," the index function will tell you what is in that position.

Uses and applications

INDEX is primarily used for looking up and referencing values from other parts of an Excel sheet or in another Excel sheet within the same workbook. As you might expect, the real value only comes in with larger datasets and Excel models. For a simple table in Excel, you do not need **INDEX** to identify the value because it is usually obvious what the cell contains. **INDEX** is often used in combination with the **MATCH** function.

The formula

=INDEX(data to be examined, row number, column number)

Errors to avoid

While it is easy to make syntax errors with **INDEX**, the bigger risk of an error is generally writing the formula correctly but putting in the wrong row or column number (or mixing up row and column in the formula). This then results in the wrong cell being referenced.

An example of the use of the **INDEX** function is as follows:

G6 =INDEX(A1:E19,8,4)

	A	B	C	D	E	F	G
1	Part ID:	Price	Cost Per Unit	Weight	Inventory on Hand		
2	A	61.00	4.27	27.00	16.00		POSITION OF PART G (DONE USING MATCH)
3	B	5.00	4.16	24.00	15.00		8
4	C	52.00	8.34	76.00	20.00		
5	D	54.00	48.99	22.00	13.00		WEIGHT OF PART G (Row 8, Column 4)
6	E	91.00	12.74	52.00	15.00		4
7	F	68.00	67.94	43.00	19.00		
8	G	40.00	28.70	4.00	18.00		
9	H	82.00	55.62	27.00	5.00		
10	I	59.00	58.81	64.00	10.00		
11	J	30.00	10.21	52.00	9.00		
12	K	48.00	26.94	42.00	12.00		
13	L	20.00	14.44	64.00	15.00		
14	M	97.00	85.00	43.00	8.00		
15	N	41.00	1.44	38.00	19.00		

Figure 2.13: *The Index function*

Conclusion

The functions shown in this chapter are among the most used and most useful of all functions in Excel. If you are going to memorize functions from just one chapter in this text, memorize these ones. They can be applied to numerous scenarios, and at least one of these functions will be used in most of the Excel models you will find in the business world.

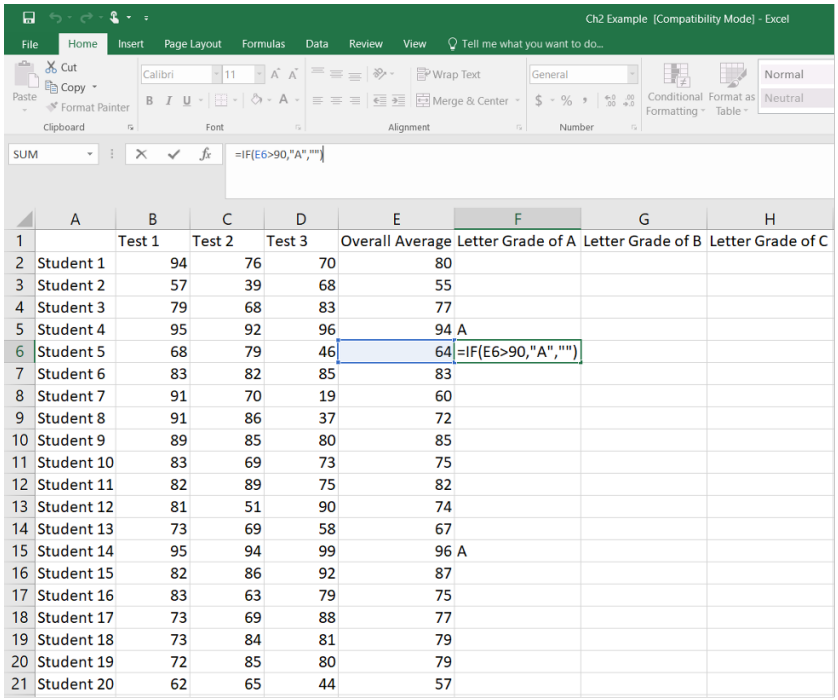
Points to remember

- **SUM** function is used for adding a group of numbers.
- IF function is used for creating If/Then type arguments in Excel.
- **LOOKUP** function is used for looking up a value from a list based upon another value.
- **VLOOKUP** function is a more advanced lookup function and the one that is most widely used.
- **MATCH** function is mainly used in combination with other functions for looking up values.
- **CHOOSE** function is mainly used in combination with the IF function to ensure the if statement meets certain criteria.
- **DATE** function is used for formatting dates in Excel given numbers for days, months, and years.

- **DAYS** function is used to find the number of days between a pair of dates.
- **FIND** and **FINDB** functions are used for finding the position of characters or strings of characters in other parts of an Excel workbook.
- **INDEX** function is used to determine what is at a location in an Excel workbook.

Multiple choice questions

1. **What value would the IF function return in *figure 2.14?***
   a. "A"
   b. "" (In other words, a blank cell is returned)
   c. "64"
   d. Not enough information as the function is not complete



	A	B	C	D	E	F	G	H
1		Test 1	Test 2	Test 3	Overall Average	Letter Grade of A	Letter Grade of B	Letter Grade of C
2	Student 1	94	76	70	80			
3	Student 2	57	39	68	55			
4	Student 3	79	68	83	77			
5	Student 4	95	92	96	94	A		
6	Student 5	68	79	46	64	=IF(E6>90,"A","")		
7	Student 6	83	82	85	83			
8	Student 7	91	70	19	60			
9	Student 8	91	86	37	72			
10	Student 9	89	85	80	85			
11	Student 10	83	69	73	75			
12	Student 11	82	89	75	82			
13	Student 12	81	51	90	74			
14	Student 13	73	69	58	67			
15	Student 14	95	94	99	96	A		
16	Student 15	82	86	92	87			
17	Student 16	83	63	79	75			
18	Student 17	73	69	88	77			
19	Student 18	73	84	81	79			
20	Student 19	72	85	80	79			
21	Student 20	62	65	44	57			

Figure 2.14: *Example query for IF function*

2. What change should be made to the following function to get it to return a value of "1"? Refer to *figure 2.15.*
   a. =Find(a,A2)
   b. =Find("A", A2)

c. =Find(1, A2)

d. =Find("p",A2)

B2 =FIND("a",A2)

	A	B
1	Address	Letter "a" position
2	Apple	#VALUE!
3	Banana	2
4	Cat	2
5	Dog	#VALUE!

Figure 2.15: *Example query for FIND function*

3. What value would be returned by the formula in *figure 2.16*?

   a. 52.00

   b. 8.34

   c. C

   d. 20.00

SUM =LOOKUP(H2,A2:E27)

	A	B	C	D	E	F	G	H
1	Part ID:	Price	Cost Per Unit	Weight	Inventory on Hand			
2	A	61.00	4.27	27.00	16.00		PART	C
3	B	5.00	4.16	24.00	15.00		PRICE	52
4	C	52.00	8.34	76.00	20.00		STOCK AVAILBLE	=LOOKUP(H2,A2:E27)
5	D	54.00	48.99	22.00	13.00			
6	E	91.00	12.74	52.00	15.00			
7	F	68.00	67.94	43.00	19.00			
8	G	40.00	28.70	4.00	18.00			
9	H	82.00	55.62	27.00	5.00			
10	I	59.00	58.81	64.00	10.00			
11	J	30.00	10.21	52.00	9.00			
12	K	48.00	26.94	42.00	12.00			
13	L	20.00	14.44	64.00	15.00			
14	M	97.00	85.00	43.00	8.00			
15	N	41.00	1.44	38.00	19.00			
16	O	96.00	79.70	75.00	20.00			
17	P	72.00	16.08	86.00	18.00			
18	Q	68.00	46.78	84.00	16.00			

Figure 2.16: *Example query for LOOKUP function*

Answers

1. **b**
2. **b**
3. **d**

Chapter 3

Logical Functions in Excel

Introduction

Logical functions are used for organizing data and creating formulas in Excel. They can also be used to analyze and compare data. In this chapter, we will cover the most popular and useful formulas you need to know to be able to use Excel effectively. Understanding these functions will let you make the most of Excel arguments and the "coding" behind more advanced Excel spreadsheets.

Structure

In this chapter, we will discuss the following topics:

- AND function
- FALSE function
- TRUE function
- IFERROR function
- IFNA function
- IFS function
- NOT function

- OR function
- XOR function
- SWITCH function

Objectives

After studying this unit, you will be able to use a variety of logical functions in Excel. These functions will enable you to build more advanced Excel models, and the formulas within those models will produce more useful results. Please note that the IF function covered in the previous chapter is also a logical function, but I do not cover it here to avoid redundancy.

AND function

The AND function enables users to check if multiple conditions are met in a spreadsheet. It is often combined with either comparison to other cells (like >, <, or =) or combined with the IF function, mentioned in *Chapter 2, The 10 Most Popular Functions in Excel.*

Uses and applications

Use the **AND** function to test if multiple conditions are met—for instance, if we wanted to assess whether a set of stores were meeting sales or revenue targets **AND** had a manager on duty for the day, and we could use the **AND** function. **AND** functions can be applied to get a subset of data that meets certain conditions—for instance, finding students who failed a test and did not submit a required homework assignment.

The formula

```
=AND(Reference Cells)
```

Figure 3.1 illustrates the use of the **AND** function:

SUM =AND(A2="Yes",IF(B2>$B$12,1,0))

	A	B	C
1	Manager on Duty?	Daily Sales	Manager And Sales Above Average
2	Yes	1240	=AND(A2="Yes",IF(B2>$B$12,1,0))
3	Yes	1350	TRUE
4	No	960	FALSE
5	Yes	1412	TRUE
6	Yes	1357	TRUE
7	No	981	FALSE
8	No	1095	FALSE
9	No	1408	FALSE
10	No	755	FALSE
11			
12	Avearge Sales	1173.111111	
13			

Figure 3.1: *The AND function*

Errors to avoid

You will likely need specific conditions to be met for each part of the **AND** function—for instance, in the preceding example, Row A must contain the word "Yes," whereas Row B must have a number higher than the average sales listed in cell B12.

FALSE function

The **FALSE** function does not have any arguments that go along with it. It is simply used to put the word false into a cell and tell other cells that reference it that the condition was not met.

Uses and applications

The **FALSE** function is very rarely used by itself—it is most likely to be used as part of another function like **IF** and return the False condition if an argument is not met.

The formula

=FALSE()

Figure 3.2 illustrates the use of the **FALSE** function:

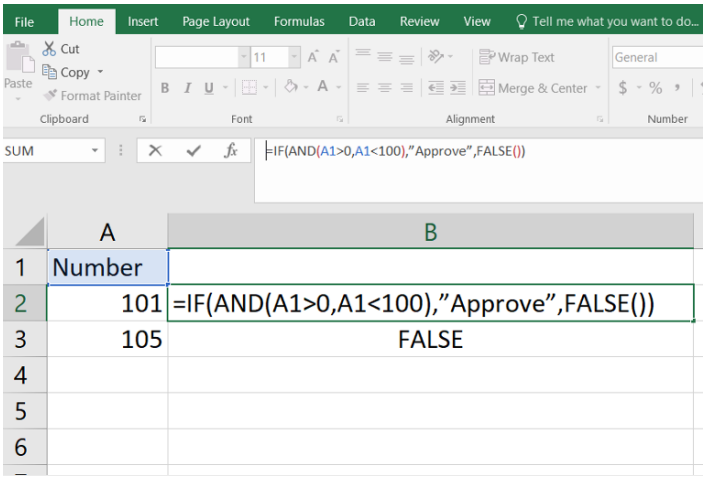


Figure 3.2: *The FALSE function*

Errors to avoid

Do not try to put any arguments into the parentheses in the **FALSE** function; the space between the "(' and ')" should not be used.

TRUE function

Like the **FALSE** function, the **TRUE** function does not have any arguments and would not be used by itself in most cases; its purpose is to be used as part of another function and to return the value True if conditions are met.

Uses and applications

The **TRUE** function is primarily used in other functions as a subset of a larger formula. It does not have any arguments of its own.

The formula

=TRUE()

Figure 3.3 illustrates the use of the **TRUE** function:

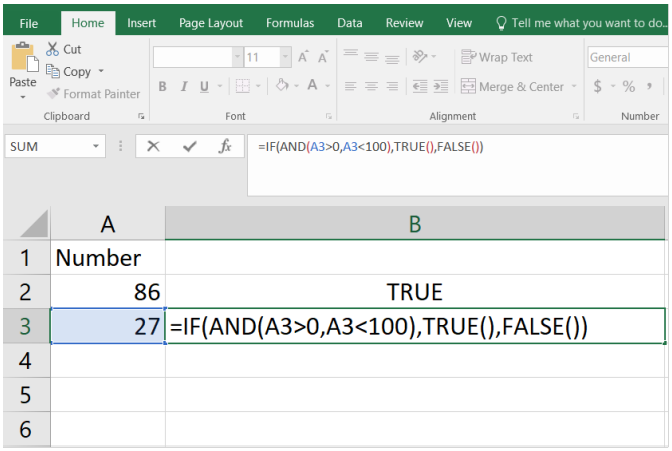


Figure 3.3: *The TRUE function*

Errors to avoid

As with the **`FALSE`** function, do not try to add any arguments to the **`TRUE`** function; it should simply be used as **`TRUE()`** in a larger formula.

IFERROR function

IF is such a popular and useful function that Excel has a series of functions related to the "If-Then" concept, and all of them are useful in slightly different ways. **`IFERROR`** is one of these.

Uses and applications

When building complex Excel models, it is quite common to have errors appear in various cells—not because you made an error in putting the sheet together necessarily, but more often because certain values in other parts of the sheet might be missing, incorrect, or unexpected. This is where **`IFERROR`** comes in; whenever an error is the result of a logical test in a formula, **`IFERROR`** returns another value instead.

The formula

=IFERROR(value, value if error occurs)

The first argument is the value to be checked for an error, and the second is the value that is returned if there is an error. The following error types evaluated: **`#N/A`**, **`#REF!`**, **`#DIV/0!`**, **`#VALUE!`**, **`#NUM!`**, **`#NAME?`**, and **`#NULL!`**.

Figure 3.4 illustrates the use of the **IFERROR** function:

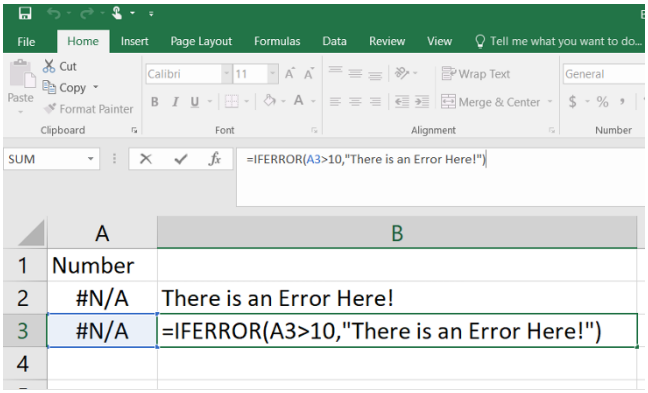


Figure 3.4: *The IFERROR function*

Errors to avoid

Be aware that in addition to the value shown, if there is an error in the cell, the **IFERROR** function also works like a regular IF statement—so if the condition being tested is true, the value **TRUE** is returned, and if the value being tested is false, the value **FALSE** is returned.

IFNA function

The IFNA function is very similar to **IFERROR**—essentially, the only difference is that **IFNA** is limited to dealing with **#N/A** errors instead of a wider variety like **IFERROR**.

Uses and applications

The uses for IFNA are functionally similar to the uses for **IFERROR**—identifying and calling out or resolving cells where there is an error or unexpected value.

The formula

=IFNA(value to be checked, result if the value being checked has the value #N/A)

Figure 3.5 illustrates the use of the **IFNA** function:

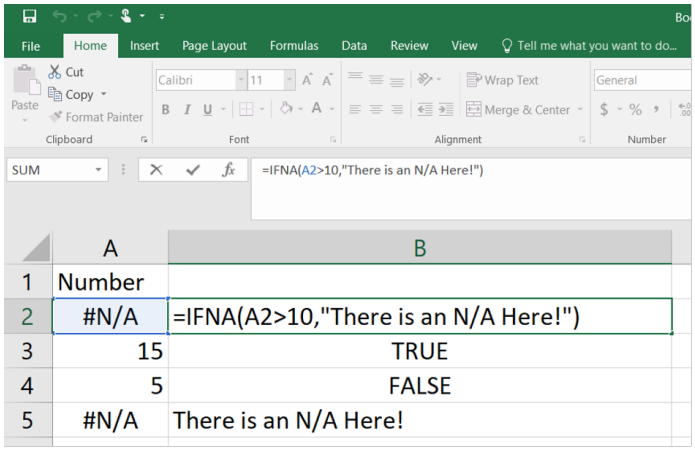


Figure 3.5: *The IFNA function*

Errors to avoid

Like **IFERROR**, **IFNA** is also an **IF** function and can have the values true and false, as the preceding figure shows. Similarly, **IFNA** can also be applied to a group or array of cells and will return an array of values itself.

IFS function

While the IF function is incredibly useful in many circumstances, oftentimes, we may need to check multiple conditions in order to determine a value to be output—you can use multiple nested IF statements to do this—or you can save time and make the sheet more readable by using the IFS function.

Uses and applications

Use the IFS function to create more advanced IF functions that encompass multiple conditions. This is very useful when building financial models—for instance, at a retail chain of stores, we might add up the daily sales for weekdays only at specific stores.

The formula

=IFS(Condition 1 to test, Value 1 if condition is met, Condition 2 to test, Value if condition 2 is met but condition 1 was NOT met)

Errors to avoid

Only newer versions of Excel will have the **IFS** function, and Macs Excel does not have it—in those cases, you will need to use nested versions of **IF**. Also, since **IFS** tests multiple conditions, it can be hard to trace the source of where any tests in the function fail. All the conditions in the **IFS** function must return either **TRUE** or **FALSE**. If it does not, the formula will give an error. If all the conditions in the **IFS** function return **FALSE**, the result of the formula would be the #N/A error. Since #N/A error is not helpful in finding out where the test failed, you can set the last Condition to **TRUE** and the value as **FALSE** or a descriptive text such as "Unknown."

NOT function

The **NOT** function is essentially like a negative sign in front of a number—it gives the opposite value of what the normal output from a cell would be.

Uses and applications

The **NOT** function is most often used in combination with other functions—it tests if a logical condition has not been met and then returns the value of true or false accordingly. A lot of what the **NOT** function does can often be accomplished with IF functions too, but **NOT** makes it a little easier at times.

The formula

=NOT(Condition to be tested)

The **NOT** cell will return a value of true or false and either display that in the cell or else use it as part of a larger formula like in the following example:

Figure 3.6 illustrates the use of the **NOT** function:

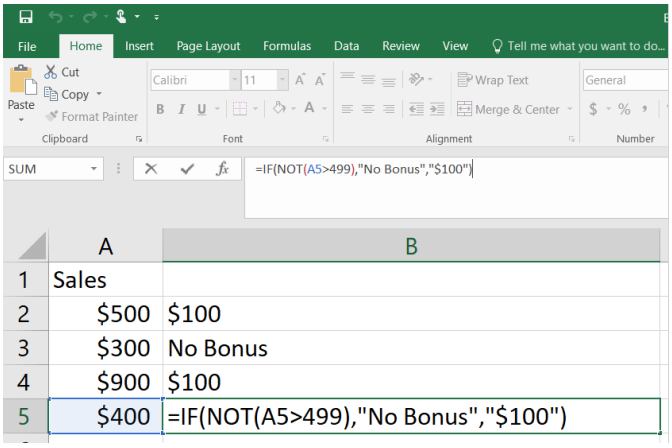


Figure 3.6: *The NOT function*

Errors to avoid

It is easy to make syntax errors with the **NOT** function—always check a few cells to see that any formulas using **NOT** are returning the expected results.

OR function

If you want to check and see if at least one of a number of conditions is met, then the OR function is the way to do it.

Uses and applications

The OR function allows you to go through a series of values and check to see if any of them are true. If they are, then the function will return the value TRUE. It can also be combined with an IF statement as in **=IF(OR(Condition 1, Condition 2, and so on.)**, value if true, value if false). This results in a much more powerful and versatile **IF** statement.

The formula

=OR(Condition 1 to be checked, Condition 2 to be checked, Condition 3, and so on.)

Figure 3.7 illustrates the use of the **OR** function:

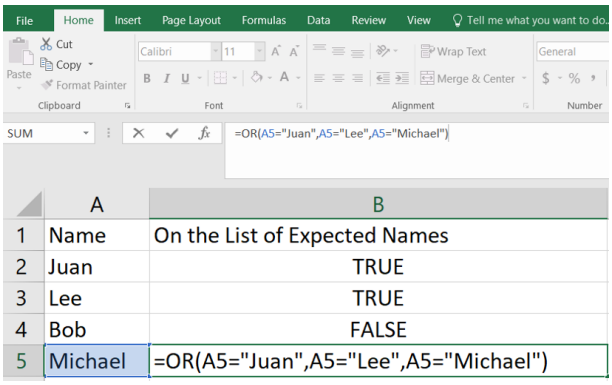


Figure 3.7: *The OR function*

Errors to avoid

Remember that the formula ignores empty cells, so if, for instance, A5 in the preceding example were blank, then the value returned would be false. This can sometimes result in a lot of "false" outputs if you are not careful about what the formula is being applied to.

XOR function

The **XOR** function, or Exclusive **OR** function, is a refinement of the standard **OR** function—**XOR** will return the value of True if one and only one of its conditions is true. If none of the conditions are true, then the value false is returned. If more than one condition is true, the value false is also returned.

Uses and applications

XOR can be useful primarily when dealing with complex logical formulas—while traditional **OR** is essentially an "inclusive OR," meaning that as long as one condition is met, it will be true, XOR requires that one and only one condition be met in order to be true. This helps when trying to categorize lists and similar tasks.

The formula

=XOR(Logical test 1, logical test 2, logical test 3, and so on.)

Errors to avoid

The most common error using **XOR** is having multiple conditions be true and returning a value of false when in fact you did not intend to require that only a single condition be true—in other words, using an **XOR** function when in fact you need an **OR** function. **XOR** is only available on Excel 2013 and the later versions.

SWITCH function

The SWITCH function tests a list of values and returns the result corresponding to the first matching value.

Uses and applications

The **SWITCH** function is a newer function for Excel, and it is primarily useful for presentation purposes. For instance, if we had a set of student grades in number form (for example, 95% or 3.8 out of 4.0), we could add another set of cells that check each student's grade and then "switch" the value out for a letter grade in its place.

The formula

=SWITCH(Value to Switch Out, Value to Match, Value to be Returned if there is a Match)

Errors to avoid

The **SWITCH** function is only found on Excel 2019 and later, so many users will not have it available to them.

Conclusion

The functions shown in this chapter are logical functions. They will primarily be used to test various logical arguments in Excel, often as part of a larger formula. With a little practice, you can use these formulas to create elegant and effective Excel sheets that can simplify many work tasks and reduce manual spreadsheet work.

Points to remember

- **AND** function: Returns TRUE if all of its arguments are TRUE.
- **FALSE** function: Returns the logical value FALSE.
- **IF** function: Specifies a logical test to perform.
- **IFERROR** function: Returns a value you specify if a formula evaluates to an error; otherwise, it returns the result of the formula.
- **IFNA** function: Returns the value you specify if the expression resolves to #N/A; otherwise, it returns the result of the expression.
- **IFS** function: Checks whether one or more conditions are met and return a value that corresponds to the first TRUE condition.
- **NOT** function: Reverses the logic of its argument.
- **OR** function: Returns TRUE if any argument is TRUE.
- **SWITCH** function: Evaluates an expression against a list of values and returns the result corresponding to the first matching value. If there is no match, an optional default value may be returned.
- **TRUE** function: Returns the logical value TRUE.
- **XOR** function: Returns a logical exclusive OR of all arguments.

Multiple choice questions

1. **How many arguments should be used in the parenthesis following the TRUE or FALSE functions?**
   a. 3
   b. 0
   c. 2
   d. 4 or more
2. **What value would the following function return if cell A2 contains "# N/A"**
   =IFERROR(A2, "Error")
   a. #N/A
   b. Error
   c. A2
   d. Not enough information
3. **What value would be returned by the following formula if BOTH condition 1 and condition 2 are true?**
   =XOR(A2>5, B2<10)
   a. True
   b. True, True
   c. False, False
   d. False

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 4

Lookup and Reference Functions in Excel

Introduction

Logical functions are used for organizing data and creating formulas in Excel. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively. Understanding these functions will let you make the most of Excel arguments and the "coding" behind more advanced Excel spreadsheets.

Structure

In this chapter, we will discuss the following topics:

- ADDRESS function
- AREAS function
- COLUMN function
- COLUMNS function
- FILTER function
- FORMULATEXT function
- GETPIVOTDATA function

- HLOOKUP function
- HYPERLINK function
- INDIRECT function
- OFFSET function
- ROW function
- ROWS function
- RTD function
- SORT function
- SORTBY function
- TRANSPOSE function
- UNIQUE function
- XLOOKUP function
- XMATCH function

Objectives

After studying this unit, you will be able to use a variety of logical functions in Excel.

ADDRESS function

The **ADDRESS** function allows the user to find the specific Excel reference, in text form, for a particular row and column number.

Uses and applications

Use the **ADDRESS** function to output reference text to a particular row and column, either with or without the **$** indicators, which make the cell an absolute value. This can be combined effectively with the **OFFSET** function.

The formula

=ADDRESS(Row Number, Column Number, Absolute Value indicator, Format, Worksheet to use)

The absolute value indicator is optional but changes the output from the formula to include or exclude anchoring/absolute cell references. The four options for absolute value are as follows:

1. Absolute
2. Absolute row; relative column
3. Relative row; absolute column
4. Relative

Figure 4.1 illustrates the use of the **ADDRESS** function:

SUM =ADDRESS(A5,B5,C5)

	A	B	C	D	E
1	Row	Column	Abs		Result
2	1	1	1		$A$1
3	1	1	2		A$1
4	1	1	3		$A1
5	1	1	4		A1
6					=ADDRESS(A5,B5,C5)
7					

Figure 4.1: *The ADDRESS function*

Errors to avoid

You should be careful to note that the output from the address function is NOT actually an Excel reference—it is just text. Thus, if you want to use the output in a separate Excel formula, for instance, summing the values found in that reference cell, you need to convert it from text to the Excel reference again.

AREAS function

The AREAS function is a lesser-used function that enables the user to determine the number of contiguous cells or a single cell.

Uses and applications

Like the other functions in this chapter, AREAS is primarily used as a reference function to establish information about another part of the sheet. It is not commonly used but can be useful for checking for errors or problems in large Excel models.

The formula

=AREAS(Reference Cells)

Note that you will need a second set of parenthesis when using the function anytime the output is bigger than 1; =AREAS([reference cells])

Figure 4.2 illustrates the use of the **AREAS** function:

9					AREAS Result and Formula	
10	A	B	C		1	AREAS(A10:B13)
11	1	2	3		2	AREAS((A10:C10,A11:C12))
12	X	Y	Z		3	AREAS((A10:C10,A11:B12,C11:C12))

Figure 4.2: *The AREAS function*

Errors to avoid

The most common error when using the AREAS function is probably forgetting to put the second set of parentheses into the formula when evaluating multiple zones or areas. *Figure 4.2* shows the formulas with double and single parentheses.

COLUMN and COLUMNS functions

The **COLUMN** and **COLUMNS** functions are both functions that return references related to the columns in a particular Excel sheet; hence, while they are different functions, we will group them together here for reference.

Uses and applications

The **COLUMN** function returns the column number for a particular cell in a worksheet, or if no cell is given, it returns the column number that the formula itself is in. The **COLUMNS** function returns the number of columns in a particular array. The two functions are most useful when combined with something like the **OFFSET** function or another reference function as part of a larger formula.

The formula

=COLUMN(Reference Cell), and =COLUMNS(Array of cells)

Figure 4.3 illustrates the use of the functions:

	A	B	C	D	E
14					
15				4	COLUMN()
16	X			1	COLUMN(A16)
17					
18	A	B	C	3	COLUMNS(A18:C18)

Figure 4.3: *The COLUMN and COLUMNS function*

Errors to avoid

COLUMN and **COLUMNS** are fairly straightforward to use, but if you are going to apply them across multiple cells, for instance, in a larger Excel model, be sure to anchor your cells accordingly.

FILTER function

The **FILTER** function is a new function added to Excel 2021. It is very useful and can be used in place of an IF statement enabling the sorting of large sets of data to pull data points that meet certain criteria.

Uses and applications

The **FILTER** function is best used for sorting large data sets based on various criteria. For instance, in *figure 4.4*, we have a list of several cities and towns, which are then categorized as either large or small. The filter function produces a list of those cities that meet the criteria as "Big."

The formula

=FILTER(Range of data to be examined or filtered, the criteria to match against a range of data, the result to be shown if no results exist).

Figure 4.4 illustrates the use of the **FILTER** function:

	A	B	C	D	E
22					
23	CITY	SIZE		Group:	Big
24	New York	Big			
25	Tokyo	Big		CITY	
26	Smallville	Small		New York	FILTER(A24:B28,B24:B28=E23,"No Results")
27	TinyTown	Small		Tokyo	
28	London	Big		London	
29					

Figure 4.4: *The filter function*

Errors to avoid

The **FILTER** function is only available in Excel 2021 and newer versions, including Excel 365.

FORMULATEXT function

FORMULATEXT is a nifty function that allows you to display the formula from another cell but as a text function.

Uses and applications

FORMULATEXT is useful for showing the specific formula being used without having to actually go to the cell in question. This makes it easy to check for errors and troubleshoot larger Excel models.

The formula

=FORMULATEXT(Cell reference for the formula you want to display)

Figure 4.5 illustrates the use of the **FOMRULATEXT** function:

	A	B	C
30			
31	Bigger		=IF(A33>2, "Bigger", "Smaller")
32			FORMULATEXT(A31)
33	3		

Figure 4.5: *The FORMULATEXT function*

Errors to avoid

This function is fairly straightforward, so you are unlikely to run into errors as long as you type the formula and use a valid reference cell.

GETPIVOTDATA function

The **GETPIVOTDATA** function lets users pull specific pieces of data from a pivot table to display.

Uses and applications

While the **GETPIVOTDATA** name may be cumbersome, the function itself is very useful—essentially, it lets you pull a specific value or piece of data out of a pivot table to display elsewhere in a sheet. This is useful if you have certain key values that you want to highlight, for instance.

The formula

=GETPIVOTDATA(The Name of the Pivot Table to pull from, a cell in the pivot table, the field/item pair to be used in pulling data)

Figure 4.6 illustrates the use of the **GETPIVOTDATA** function:

36	**Row Label**	**Sum of Product A**	**Sum of Product B**	**Sum of Product C**
37	China	195	200	340
38	Germany	160	310	725
39	India	85	450	525
40	United State	100	300	650
41	**Grand Total**	**540**	**1260**	**2240**
42				
43	REGION	Germany		
44	PRODUCT	Product A		
45	SALES	540		
46		=GETPIVOTDATA("Sum of Product A",$B$37:$D$40)		

Figure 4.6: *The GETPIVOTDATA function*

Errors to avoid

The function is fairly straightforward to use, but it is fairly common to make a mistake when putting in the field names in the formula, so be sure to check those carefully.

HLOOKUP function

The **HLOOKUP** function finds and displays a value from data in a horizontal table. The "H" in **HLOOKUP** stands for "horizontal." Importantly, the lookup values must be in the first row of the table, moving horizontally to the right.

Uses and applications

HLOOKUP is not as widely used as **VLOOKUP** simply because tables are more often vertically displayed than horizontally, but it is still a pretty useful function for a lot of circumstances. Like **VLOOKUP**, **HLOOKUP** works with approximate and exact matching and wildcards (* and ?) for finding partial matches.

The formula

=HLOOKUP(Value to look up, table of cells to look for the value, Row to display from the table where value is found, Exact/Approximate indicator)

Figure 4.7 illustrates the use of the **HLOOKUP** function:

	A	B	C	D	E	F	
35							
36	PRODUCT	Product A	Product B	Product C		US Sales	
37	United States	100	300	650		300	Product B
38	India	85	450	525		=HLOOKUP(G37,A36:D40,2,0)	
39	China	195	200	340			
40	Germany	160	310	725			
41							

Figure 4.7: *The HLOOKUP function*

Errors to avoid

Similar to **VLOOKUP**, where the value you are looking up must be the first column of the table, with **HLOOKUP**, the value you are looking up must be in the first row of the table. For instance, if the formula for **HLOOKUP** used in the preceding example were modified so it started at row 35 instead of row 36 (for example, A35:D40), the formula would produce an error (#N/A) result.

HYPERLINK function

As the name implies, the **HYPERLINK** function creates a hyperlink in a cell from a specified destination and link text.

Uses and applications

The **HYPERLINK** function is great for creating Excel models that reference outside resources or data. You can use the function to set a clickable hyperlink to other workbook locations, pages on the internet, or even files on network servers. This makes the function very useful for models that deal with large datasets that are handled in separate workbooks or files.

The Formula

=HYPERLINK(the link address, the link text to display in the cell)

Figure 4.8 illustrates the use of the **HYPERLINK** function:

	A
49	
50	https://bpbonline.com/
51	
52	Buy Technology Books Here
53	
54	=HYPERLINK(A50, "Buy Technology Books Here")

Figure 4.8: *The AREAS function*

Errors to avoid

This function is fairly straightforward, so you are unlikely to run into errors as long as you type the formula and use a valid reference cell.

INDIRECT function

The **`INDIRECT`** function is one of those functions that does not seem like it would be useful but is often critically important if you want to build a complex dynamic Excel model. Basically, **`INDIRECT`** creates a valid cell reference from a starting from a text string.

Uses and applications

`INDIRECT` is useful when you want to create a text value that can be used as a valid reference for a formula in Excel. As a result, **`INDIRECT`** will pair with **`ADDRESS`** and similar Excel functions.

The formula

=INDIRECT(Reference Cells)

Figure 4.9 illustrates the use of the **`INDIRECT`** function. Notice that we have built a formula that pulls the same cell from three different worksheets in the workbook. Imagine that we had a large Excel file with 25 different worksheets, all displaying different information—perhaps related to a particular client account. We might want a summary page that displays the customer account balance for all of those worksheets without having to go to each individual worksheet. This is where **`INDIRECT`** is useful. And the nice feature about the function versus simply using **"`=Sheet1!A58`"** is that we can see what cells we are linking to and update them quickly and easily as needed. This can also enable us to build dropdowns, buttons,

and other more graphically friendly options into Excel models. **INDIRECT** is one of my favorite Excel functions for this reason.

	A	B	C	D
55				
56	**ShOw ThIs VaLuE**			
57	Sheet	Cell	Value	
58	Sheet1	A56	ShOw ThIs VaLuE	=INDIRECT(A58&"!"&B58)
59	Sheet2	A56	ShEeT2	
60	Sheet3	A56	sHeEt3	
61				
62				
63				
64				
65				
66				
67				
68				
69				
70				
71				
72				

Sheet1 Sheet2 Sheet3

Figure 4.9: *The INDIRECT function*

Errors to avoid

The most common error when using the **INDIRECT** function is probably forgetting to add the "!" mark after a sheet reference and before the cell reference.

OFFSET function

The OFFSET function creates a reference to another cell and returns the value in that cell. To use OFFSET, you need five inputs:

1. a starting point
2. a row offset
3. a column offset
4. a height in rows
5. a width in columns

Uses and applications

Like the other functions in this chapter, **AREAS** is primarily used to pull information from another part of the sheet, and as a result, it is handy in formulas that require a dynamic range, especially when paired with the **COLUMN** or **COLUMNS** function and the **ROW** or **ROWS** function.

The formula

=OFFSET (The starting point in the sheet, Number of rows to offset below the starting point, Number of columns to offset to the right of the starting reference, [optional] The height in rows of the returned reference, [optional] The width in columns of the returned reference)

Figure 4.10 illustrates the use of the **OFFSET** function:

	A	B	C
65			
66	ShOw ThIs VaLuE		
67	*=OFFSET(B67,2,1,1,1)*	Starting Here	Not what we are looking for
68		Not this value	This is not the cell we want
69		Don't show this cell	**ShOw ThIs VaLuE**

Figure 4.10: *The OFFSET function*

Errors to avoid

Using **OFFSET**, you are most likely to run into problems simply with having miscounted the number of rows or columns you want to move from the original reference cell—be sure to double-check this in the formula.

ROW and ROWS functions

The **ROW** and **ROWS** functions are very similar to the **COLUMN** and **COLUMNS** functions. **ROW** tells you the row number of a specific cell in a sheet, whereas ROWS tells you the number of **ROWS** within a specific range.

Uses and applications

ROW and **ROWS** can be used to good effect when combined with other functions. For instance, we could combine **ROW** and **OFFSET** or **ROW** and **INDIRECT** to build a dynamic formula that would return a specific value from a value based on inputs from the user.

The formula

=ROW(Reference cell where the row is to be looked up)

=ROWS(Range of cells to look up the number of rows)

Figure 4.11 illustrates the use of the **ROW** and **ROWS** functions:

	A	B	C
72			
73			
74	77	=ROW(A77)	
75	6	=ROWS(C75:C80)	How
76			Many
77	What Row is this?		Rows
78			In
79			This
80			Range?

Figure 4.11: *The ROWS and ROW functions*

Errors to avoid

You are unlikely to make an error in the **ROW**, or **ROWS** functions themselves, but when combining these functions into a more complex formula, such as with **INDIRECT**, it is easy to make a syntax error. Be sure to double-check that the result you are looking for is what the function produces.

RTD function

The RTD function is a very useful but underappreciated one. **RTD** stands for **Real Time Data**, and the function pulls data from a server that carries information being updated over time.

Uses and applications

It is easy to imagine uses for RTD but harder to show them because the function pulls from outside Excel. As an example of the RTD function, imagine pulling information from a server that delivers continuous data for stock market prices. Assume that the server is provided as a **Dynamic Link Library** (**DLL**) named StockPriceReport.dll, that it is installed on the user's computer, and that it is registered as a COM add-in by its own setup process. The RTD function can be used as part of **user-defined functions** (**UDFs**) in Microsoft Visual Basic for Applications. These functions then serve as wrappers for the RTD function, letting users focus only on the data that is important to them.

The formula

=RTD(RealTimeServerProgID,ServerName,Topic1,[Topic2], ...)

Here, the **`RealTimeServerProgID`** is the string name of the **`ProgramID`** to be pulled from the server, **`ServerName`** is the name of the server, and the Topic numbers (only 1 of which is required) are the strings that determine the data point to be pulled.

In the scenario we outline preceding, the general form of the RTD function as used with this server could be the following:

=RTD("StockPriceProdID","StockServer","StockID","StockPrice","PriceTime")

Errors to avoid

RTD is a very useful function, as you can probably imagine, but because it pulls from outside Excel, it is also easy to make errors—errors related to file names, variable names, server names, and so on—and to have unexpected results or missing values if the server is not available, there is an issue with data on the server or a similar issue. As a result, while the function can certainly create some very useful Excel models, it is one to use with caution.

SORT and SORTBY functions

The **`SORT`** and **`SORTBY`** functions let you sort lists in Excel based on various characteristics. Sorting in Excel is something that has been possible almost since the inception of Excel by using the menu bar, but Excel 2021 introduced **`SORT`** as a function, making it easier to sort data quickly and efficiently.

Uses and applications

The **`SORT`** function sorts a set of data based on a column within the array of data. In contrast, **`SORTBY`** lets you produce a sorted set of data based on a column within the array. Both functions are useful—use **`SORT`** for organizing a dataset and **`SORTBY`** to pull sorted lists from the data.

The formula

=SORT(Range or array of cells to sort, Column index to use for sorting, Sorting Index, By Column)

Note that for the column to sort by, the Default is 1. This is an optional argument in Excel.

Sorting order is also optional, and 1 = Ascending, –1 = Descending. The Default is in ascending order.

By_col is optional again, with TRUE = sort by column. FALSE = sort by row. Default is FALSE.

Figure 4.12 illustrates the use of the SORT function, sorting the firms by the number of employees in ascending order:

	A	B	C	D
83				
84	Employee Group	Number of Employees		(SORT(A84:B89,2))
85	A	10		
86	D	14		
87	C	16		
88	E	18		
89	B	22		
90				

Figure 4.12: *The SORT function*

Errors to avoid

SORT and **SORTBY** are only available in Excel 2021 and later versions, so everyone will not have these features. However, if you do not have the formula, you should still at least have the sorting option through the Excel menu system.

TRANSPOSE function

The **TRANSPOSE** function flips a set of data on its axis; a set of vertical data in a series of rows becomes horizontal data in columns, and a set of horizontal data in columns becomes vertical data in rows.

Uses and applications

TRANSPOSE is very useful for taking a set of raw data that might not be in the shape or structure you want it in and then turning it into part of a larger dataset. For instance, we might have a series of sales projections in a calendar format spread out over a set of horizontal cells. If we want to turn that horizontal set of cells into a vertical set of rows, **TRANSPOSE** will let us do that.

The **TRANSPOSE** function is useful if you need a dynamic approach that will update the output cells when source data changes, but if you only need a one-time transposition, it is probably faster to use Paste Special with the Transpose option.

The formula

`=TRANSPOSE(Reference Cells)`—note, however, this is an array function. In order to use the array function, you must follow the following steps:

1. Select a set of blank cells equal to the number of cells you want to transpose but running in the opposite direction.
2. With those blank cells still selected, type **`=TRANSPOSE( Reference Cells)`**
3. When finished, do NOT press ENTER. Instead, you need to press CONTROL, SHIFT, and ENTER all at the same time. To make this easier, hold down CONTROL and SHIFT, then press the ENTER key.

Figure 4.13 illustrates the use of the **`TRANSPOSE`** function:

	A	B	C	D	E
90					
91	Month	January	February	March	April
92	Sales	100	105	120	110
93					
94					
95	Month	Sales		{=TRANSPOSE(A91:E92)}	
96	January	100			
97	February	105			
98	March	120			
99	April	110			

Figure 4.13: *The TRANSPOSE function*

Errors to avoid

The **`TRANSPOSE`** function is an array function, which means it is harder to use than a typical Excel function. In order to get it to work properly, you must follow the preceding outlined steps in the formula section.

UNIQUE function

If you have ever had a list of something in Excel—names, dates, numbers, and so on—and wanted to remove the duplicates, **`UNIQUE`** is the function you want to use.

Uses and applications

The **`UNIQUE`** function is new for Excel 2021 / Excel 365, but it is a very useful addition to the software. **`UNIQUE`** lets you pull a list of data points that occur once and thus are "unique" from a larger data set. This is very useful for weeding out duplicate data,

especially if combined with a **VLOOKUP** function to create a data subset of unique values and associated data points with those values.

The formula

=UNIQUE (Reference Cells, [by_col], [exactly_once]).

Here,

Reference cells: range or array from which to extract unique values.

by_col: Optional value to let you choose whether to draw unique values by columns versus rows. By row = FALSE (Default); by column = TRUE.

exactly_once: Optional where TRUE = values that occur once, FALSE= all unique values (default).

Errors to avoid

The **UNIQUE** function is only available in Microsoft's Excel 365 (2021 version), so some users will not have access to it.

XLOOKUP function

Excel **XLOOKUP** function is designed as a replacement for older functions, including **VLOOKUP**, **HLOOKUP**, and **LOOKUP**. It is only available in Microsoft's Excel 365 (2021 version) and (presumably) future versions.

Uses and applications

The big difference for the new function is that **XLOOKUP** works in multiple directions (vertical and horizontal), which means you do not need a different function depending on which direction you are looking at. Like the older functions, **XLOOKUP** supports approximate and exact matching and wildcards (* ?) for partial matches.

The formula

=XLOOKUP (Lookup, Lookup_array, Return_array, [not_found], [match_mode], [search_mode])

For **XLOOKUP**, the component arguments are as follows:

Lookup: The lookup value.

Lookup_array: The array or range to search.

`Return_array`: The array or range to return.

`[Not_found]`: Optional and gives a value to return if no match found.

`[Match_mode]`: Optional where 0 = exact match (default), –1 = exact match or next smallest, 1 = exact match or next larger, 2 = wildcard match.

`[Search_mode]`: Optional where 1 = search from first (default), –1 = search from last, 2 = binary search ascending, –2 = binary search descending.

Errors to avoid

`XLOOKUP` is only available in Excel 365, so if you do not have it, use **`HLOOKUP`** and **`VLOOKUP`** instead.

XMATCH function

`XMATCH` performs a lookup and returns a position in vertical or horizontal ranges.

Uses and applications

The **`XMATCH`** function is intended as a replacement for the original **`MATCH`** function. **`XMATCH`** supports approximate and exact matching, reverse search, and wildcards (* ?) for partial matches.

The formula

=XMATCH (Lookup_value, Lookup_array, [Match_mode], [Search_mode])

`Lookup_value`: The value to be looked up

`Lookup_array`: The dataset or range to search for the Lookup Value

`Match_mode`: An optional value where 0 = exact match (default), –1 = exact match or next smallest, 1 = exact match or next larger, 2 = wildcard match.

`Search_mode`: Another optional argument where 1 = search from first (default), –1 = search from last, 2 = binary search ascending, –2 = binary search descending.

Errors to avoid

As with **`XLOOKUP`**, **`XMATCH`** is only available in Excel 365 (the Excel 2021 version).

Conclusion

The functions shown in this chapter are reference and lookup functions, which can be very helpful for creating datasets, creating relevant results summaries from large pools of data, and building Excel models that pull from multiple data sources. Several of these functions, as noted, are only available in the cloud-based Excel 365, but even if you do not have that program, many of the other functions can make your life much easier when you are building Excel models.

Points to remember

- **ADDRESS** function—returns a reference as text to a single cell in a worksheet.
- **AREAS** function—returns the number of areas in a reference.
- **COLUMN** function—returns the column number of a reference.
- **COLUMNS** function—returns the number of columns in a reference.
- **FILTER** function—filters a range of data based on criteria you define.
- **FORMULATEXT** function—returns the formula at the given reference as text.
- **GETPIVOTDATA** function—returns data stored in a PivotTable report.
- **HLOOKUP** function—looks in the top row of an array and returns the value of the indicated cell.
- **HYPERLINK** function—creates a shortcut that opens a document stored on a network server, an intranet, or the internet.
- **INDIRECT** function—returns a reference indicated by a text value.
- **OFFSET** function—returns a reference offset from a given reference.
- **ROW** function—returns the row number of a reference.
- **ROWS** function—returns the number of rows in a reference.
- **RTD** function—retrieves real-time data from a program that supports COM automation
- **SORT** function—sorts the contents of a range or array.
- **SORTBY** function—sorts the contents of a range or array based on the values in a corresponding range or array.
- **TRANSPOSE** function—returns the transpose of an array.
- **UNIQUE** function—returns a list of unique values in a list or range.

- **`XLOOKUP`** function—searches a range or an array and returns an item corresponding to the first match it finds.
- **`XMATCH`** function—returns the relative position of an item in an array or range of cells.

Multiple choice questions

1. **What function is intended as a replacement for VLOOKUP and HLOOKUP?**
   a. XMATCH
   b. XLOOKUP
   c. LOOKUP
   d. All of the above

2. **What buttons must be pushed after entering the TRANSPOSE function?**
   a. Control, Alt, Delete
   b. Control, Shift, and Enter
   c. Enter
   d. Shift and Enter

3. **If you wanted to display a formula you had used in Excel without having to mouse over the cell where the formula was entered, what function could you use?**
   a. =CHOOSE()
   b. =LOOKUP( )
   c. =RTD( )
   d. =FORMULATEXT( )

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 5

Math Functions in Excel

Introduction

Excel is filled with mathematical functions that can make any virtual computation much easier. This is an important task for students and workers alike. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- ABS function
- ACOS and ACOSH functions
- ACOT and ACOTH functions
- ASIN and ASINH functions
- ATAN and ATANH functions
- CEILING and FLOOR functions
- COMBIN and COMBINA functions

- COS and COSH functions
- COT and COTH Functions
- CSC and CSCH functions
- DECIMAL function
- DEGREES and RADIANS functions
- EVEN and ODD functions
- INT function
- LCM function
- LN and LOG function
- MMULT function
- MOD function
- POWER function
- PRODUCT function
- RAND and RANDBETWEEN functions
- ROMAN function
- ROUND, ROUNDUP, and ROUNDDOWN functions
- SEC and SECH functions
- SIGN function
- SIN and SINH function
- SQRT function
- SUBTOTAL function
- SUMIF, SUMPRODUCT, and SUMSQ function
- TAN and TANH function

Objectives

After studying this unit, you will be able to apply the most common math functions in Excel and determine the appropriate circumstances where these formulas can be effectively applied.

ABS function

The **ABS** function is a simple Excel function that gives the absolute value of a particular number in Excel. This is useful if you want to compare magnitudes of figures while ignoring the sign on the figure's value.

Uses and applications

Use the **ABS** function to get the absolute value of an amount. The positive values will be unchanged, whereas negative values will have the sign flipped to positive.

The formula

=ABS(Reference Cells)

Figure 5.1 illustrates the use of the ABS function:

	A	B	C	D
1	Location	Expected Change in Sales	Abs. Change in Sales	
2	A	-8%	8.00%	=ABS(B2)
3	B	5%	5.00%	
4	C	10%	10.00%	
5	D	-12%	12.00%	

Figure 5.1: *The ABS function*

Errors to avoid

The **ABS** function is straightforward to use. You just have to make sure to reference the intended cells, and you will not have any problems.

ACOS and ACOSH functions

The **ACOS** and **ACOSH** functions are part of a family of trigonometric functions available in Excel. They give Excel users the Arccosine or inverse of the cosine function and the inverse of the hyperbolic cosine, respectively.

Uses and applications

ACOS and **ACOSH** are fairly specialized functions. They are most likely to be used in engineering or math applications where a trigonometric calculator is needed. The advantage of doing the calculation in Excel versus hard coding the values is that the

Excel sheet becomes dynamic. The value returned in the calculation is in radians, so you can convert it to degrees using the **DEGREES** function.

The formula

=ACOS(Reference Cells)

or

=ACOSH(Reference Cells)

Figure 5.2 illustrates the use of the **ACOS** and **ACOSH** functions:

7	Value	Arccosine	Inverse Hyperbolic Cosine
8	-1.00	3.14	1.81
9	-0.50	2.09	1.37
10	0.00	1.57	1.02
11	0.50	1.05	0.31
12	1.00	0.00	#NUM!
13		Range: -1 to 1	Range: Greater than 0
14		=ACOS(A12)	=ACOSH(B12)

Figure 5.2: *The ACOS and ACOSH functions*

Errors to avoid

The **ACOS** and **ACOSH** functions themselves are straightforward to use. However, the issue one might run into is in using an invalid value based on misunderstanding the calculation being done by the formula. For instance, the Arccosine will only return valid numbers for reference cells, which are numbers in the range of –1 to +1. Similarly, **ACOSH** requires a number larger than 0. Thus, as the preceding figure shows, the **ACOSH** function referring to 0 returns the value: **#NUM!**

ACOT and ACOTH functions

The ACOT and ACOTH functions are another part of a family of trigonometric functions available in Excel. They give Excel users the Arccotangent or inverse of the cotangent function and the inverse of the hyperbolic cotangent, respectively.

Uses and applications

Again, **ACOT** and **ACOTH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. The advantage of doing the calculation in Excel over hard coding the values is that the

Excel sheet becomes dynamic. Again, the values returned are in radians rather than degrees for the **ACOT** function.

The formula

=ACOT(Reference Cells)

Or

=ACOTH(Reference Cells)

Figure 5.3 illustrates the use of the ACOT and ACOTH functions:

16	Value	Arccotangent	Inverse Hyperbolic Cotangent
17	-10.00	3.04	0.34
18	-5.00	2.94	0.35
19	0.00	1.57	0.75
20	0.64	1.00	5.16
21	1.00	0.79	#NUM!
22		Range: Any real number	Range: Abs. value of number greater than 1.0
23		=ACOT(A21)	=ACOTH(B21)
24			

Figure 5.3: *The ACOT and ACOTH functions*

Errors to avoid

ACOT and **ACOTH** are simple functions to use, provided that the values used in the calculation are valid. Arccotangents can be calculated based on any real number, and inverse hyperbolic cotangents can also be calculated as long as the number has an absolute value in excess of 1.0.

ASIN and ASINH functions

The **ASIN** and **ASINH** functions are another part of a family of trigonometric functions available in Excel. They give Excel users the arcsine or inverse of the sine function and the inverse of the hyperbolic sine, respectively.

Uses and applications

Again, **ASIN** and **ASINH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. The advantage of doing the calculation in Excel, hard coding the values, is that the Excel sheet becomes dynamic. Again, the values returned are in radians rather than degrees for the ASIN function.

The formula

=ASIN(Reference Cells) and ASINH(Reference Cells)

Figure 5.4 illustrates the use of the **ASIN** and **ASINH** functions:

25	Value	Arcsine	Inverse Hyperbolic Sine
26	-1.00	-1.57	-1.23
27	-0.50	-0.52	-0.50
28	0.00	0.00	0.00
29	0.50	0.52	0.50
30	1.00	1.57	1.23
31		Range: -1 to 1	Range: Any real number
32		=ASIN(A30)	=ASINH(B30)

Figure 5.4: *The ASIN and ASINH functions*

Errors to avoid

ASIN and **ASINH** are simple functions to use, provided that the values used in the calculation are valid. Arcsines can be calculated for values in the –1 to +1 range, and inverse hyperbolic sines can be calculated for any number in Excel.

ATAN and ATANH functions

Like some other functions that we have discussed, **ATAN** and **ATANH** functions are another part of a family of trigonometric functions available in Excel. They give Excel users the arctangent or inverse of the tangent function and the inverse of the hyperbolic tangent, respectively.

Uses and applications

As with other trigonometric functions, **ATAN** and **ATANH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. The advantage of doing the calculation in Excel over hard coding the values is that the Excel sheet becomes dynamic. Again, the values returned are in radians rather than degrees for the **ATAN** function.

The formula

=ATAN(Reference Cells)

Or

=ATANH(Reference Cells)

Figure 5.5 illustrates the use of the **ATAN** and **ATANH** functions:

35	Value	Arctangent	Inverse Hyperbolic Tangent
36	-1.55	-1.00	-3.41
37	-0.50	-0.46	-0.50
38	0.00	0.00	0.00
39	0.50	0.46	0.50
40	10.00	1.47	#NUM!
41		Range:Any real number	Range: -1 to +1
42		=ATAN(A40)	=ATANH(B40)

Figure 5.5: *The ATAN and ATANH functions*

Errors to avoid

ATAN and **ATNH** are simple functions to use, provided the values used in the calculation are valid. Arctangents can be calculated for any real number, and the inverse hyperbolic sines can be calculated for values in the –1 to +1 range. When a number is outside that range, the function will return #NUM!, as shown preceding.

CEILING and FLOOR functions

The **CEILING** function is a handy tool for rounding numbers up to the nearest unit of value, while the **FLOOR** function rounds values down to a given unit.

Uses and applications

CEILING and **FLOOR** are most useful when you want to change a number to match a given increment. For instance, these can be used in a scenario where if you had prices originally set in 0.01-unit increments but wanted to round everything up (for **CEILING**) or down (for **FLOOR**) to the nearest whole unit of currency.

The formula

=CEILING(Reference Cell to be rounded, increment to round to)

or

=FLOOR(Reference Cell to be rounded, increment to round to)

Figure 5.6 illustrates the use of the **CEILING** and **FLOOR** functions:

64	Original Value	Rounded Value	Formula
65	0.00	0.00	=CEILING(A65,0.1)
66	0.11	0.20	=CEILING(A66,0.1)
67	0.37	0.50	=CEILING(A67,0.5)
68	0.54	1.00	=CEILING(A68,1)
69	1.03	2.00	=CEILING(A69,2)
70	0.00	0.00	=FLOOR(A70,0.1)
71	0.11	0.10	=FLOOR(A71,0.1)
72	0.77	0.50	=FLOOR(A72,0.5)
73	1.03	1.00	=FLOOR(A73,1)
74	1.99	0.00	=FLOOR(A74,2)

Figure 5.6: *The CEILING and FLOOR functions*

Errors to avoid

Remember that the **CEILING** function always rounds up to a given number of decimals, and the **FLOOR** function always rounds down. So, for instance, the **FLOOR** output for 0.99 rounded to 1.0 will be 0, and the **CEILING** output for 0.01 rounded to 1.0 will be 1. It is common to make errors in setting the unit to be rounded, so it is necessary to check that value carefully.

COMBIN and COMBINA functions

The **COMBIN** and **COMBINA** functions are calculation functions to determine the number of potential combinations of items given a chosen set. For instance, if we have a set of 10 letters and want to use three of them in some combination, such as ABC, FDB, and so on, the **COMBIN** function will tell us the number of possible options for combinations. **COMBINA** calculates combinations with repetition among values (ABC, ACB, and so on), whereas **COMBIN** does not allow repetition in values.

Uses and applications

COMBIN and **COMBINA** are handy for doing calculations in place of using a calculator. The functions are most likely to be used in probability applications such as insurance.

The formula

=*COMBIN(Number of Items to Choose from, Number of Items in each combination)*

or

=*COMBINA(Number of Items to Choose from, Number of Items in each combination).*

The difference between the two functions is in the calculation done. **COMBINA** allows replacement or repetition of values, whereas **COMBIN** does not, meaning that **COMBINA** will always be at least as large as **COMBIN**.

Figure 5.7 illustrates the use of the **COMBIN** and **COMBINA** functions:

76	# of Items	# Chosen	Combinations	Combinations with Repetition
77	10.00	1	10	10
78	10.00	3	120	220
79	20.00	5	15504	42504
80			=COMBIN(A79,B79)	=COMBINA(A79,B79)

Figure 5.7: *The COMBIN and COMBINA functions*

Errors to avoid

The easiest error to make when using **COMBIN** and **COMBINA** is mixing the two formulas up; using **COMBIN** instead of **COMBINA**, for instance. Be sure you understand your application and whether the replacement of items chosen is being used in the situation.

COS and COSH functions

The **COS** and **COSH** functions are another part of a family of trigonometric functions available in Excel. They give Excel users the cosine function and the hyperbolic cosine, respectively.

Uses and applications

Again, **COS** and **COSH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. Like before, the advantage of doing the calculation in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, so if the data are in degrees instead, the function **RADIANS** should be used.

The formula

=COS(Reference Cells)

Or

=COSH(Reference Cells)

Figure 5.8 illustrates the use of the **COS** and **COSH** functions:

	A	B	C
16	Value	Cosine	Hyperbolic Cosine
17	-3.14	-1.00	1.54
18	-1.00	0.54	1.15
19	0.00	1.00	1.54
20	1.00	0.54	1.15
21	3.14	-1.00	1.54
22		Input: Angle in Radians	Range: Any real number
23		=COS(A21)	=COSH(B21)

Figure 5.8: *The COS and COSH functions*

Errors to avoid

The easiest error to make in the **COS** function is inputting degrees instead of radians. So be careful regarding the input values.

COT and COTH functions

The **COT** and **COTH** functions are another part of a family of trigonometric functions available in Excel. They give Excel users the cotangent function and the hyperbolic cotangent, respectively.

Uses and applications

Again, **COT** and **COTH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. Like before, the advantage of doing the calculation in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, and so, if the data is in degrees instead, the function RADIANS should be used.

The formula

=COT(Reference Cells)

Or

=COTH(Reference Cells)

Figure 5.9 illustrates the use of the **COT** and **COTH** functions:

45	Value	Cotangent	Hyperbolic Cotangent
46	-3.14	627.88	1.00
47	-1.00	-0.64	-1.77
48	0.00	#DIV/0!	#DIV/0!
49	1.00	0.64	1.77
50	3.14	-627.88	-1.00
51		Input: For COT and COTH must be smaller than ABS value of 2 ^27	
52		=COT(A50)	=COTH(B50)

Figure 5.9: *The COT and COTH functions*

Errors to avoid

The easiest error to make in the COT function is inputting degrees instead of radians. Be careful about the input values. The range for both COT and COTH is any number with an absolute value smaller than 2 to the 27th power—or roughly 134.2 million.

CSC and CSCH functions

The CSC and CSCH functions are also part of a family of trigonometric functions available in Excel. They give Excel users the cosecant function and the hyperbolic cosecant, respectively.

Uses and applications

Again, **CSC** and **CSCH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. Like before, the advantage of doing the calculation in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, and so if the data are in degrees instead, the function RADIANS should be used.

The formula

=CSC(Reference Cells)

Or

=CSCH(Reference Cells)

Figure 5.10 illustrates the use of the **CSC** and **CSCH** functions:

55	Value	Cosecant	Hyperbolic Cosecant
56	-3.14	-627.88	0.00
57	-1.00	-1.19	-0.67
58	0.00	#DIV/0!	#DIV/0!
59	1.00	1.19	0.67
60	3.14	627.88	0.00
61		Input: Angle in Radians	Range: Any real number less than ABS (2 ^27)
62		=CSC(A60)	=CSCH(B60)

Figure 5.10: *The COT and COTH functions*

Errors to avoid

The easiest error to make in the CSC function is inputting degrees instead of radians. Be careful about the input values. The range for both CSC and CSCH is any number with an absolute value smaller than 2 to the 27th power—or roughly 134.2 million.

DECIMAL function

The **DECIMAL** function is a calculation function primarily used to switch between different number bases. For instance, **DECIMAL** can be used to convert from Hexadecimal (base 16) values to standard base 10 values.

Uses and applications

DECIMAL is most useful when taking a list of values in one base system, such as base 8 or base 16, and switching them to standard values. This is helpful in niche engineering and math applications.

The formula

=DECIMAL(Text value to be converted, base number system on text value)

Figure 5.11 illustrates the use of the **DECIMAL** function:

82	Hexademical Value	Number Base	Output
83	FF	16	255
84			=DECIMAL(A83,B83)

Figure 5.11: *The DECIMAL function*

Errors to avoid

Note that the initial value to be converted in the **DECIMAL** function must be in text format, and the next argument must specify the number base on this text value. The **DECIMAL** function can be tricky, so be sure to double-check any values if the results seem unexpected.

DEGREES and RADIANS functions

The **DEGREES** and **RADIANS** functions convert angles measured in degrees to radians and vice-versa. These functions pair nearly with the family of trigonometric functions available in Excel.

Uses and applications

The **DEGREES** and **RADIANS** functions are quite useful for switching between the two primary measures for angles, either for use with trigonometric functions or for use with a list of values that need to be standardized to one type or the other.

The formula

=DEGREES(Angle to be converted to degrees)

Or

=RADIANS(Angle to be converted to radians)

Figure 5.12 illustrates the use of the **DEGREES** and **RADIANS** functions:

86	Degrees	Radians (Formula)	Degrees (Formula)
87	0	0.00	0
88	45	0.79	45
89	90	1.57	90
90	135	2.36	135
91	180	3.14	180
92	225	3.93	225
93	270	4.71	270
94	315	5.50	315
95	359	6.27	359
96		=RADIANS(A95)	=DEGREES(B95)

Figure 5.12: *The COT and COTH functions*

Errors to avoid

The **DEGREES** and **RADIANS** functions are fairly straightforward to use—just be sure to include the "S" at the end of each word in the formula.

EVEN and ODD functions

The **EVEN** and **ODD** functions are both specialties rounding functions. Both functions round positive numbers up and negative numbers down, to the nearest integer, with EVEN rounding to the nearest even number and **ODD** rounding to the nearest odd number.

Uses and applications

The **EVEN** and **ODD** functions are primarily useful when dealing with a large set of numbers that need to be rounded to either an even or odd value; for instance, for the categorization of values in groups.

The formula

=EVEN(Value to be rounded)

Or

=ODD(Value to be rounded)

Figure 5.13 illustrates the use of the **EVEN** and **ODD** functions:

98	Value	Even	Odd
99	-9.1	-10	-11
100	-1.6	-2	-3
101	0.3	2	1
102	2.2	4	3
103	3.8	4	5
104		=EVEN(A103)	=ODD(A103)

Figure 5.13: *The ODD and EVEN functions*

Errors to avoid

The **ODD** and **EVEN** functions are simple formulas, but remember that negative numbers are rounded down, and positive numbers are rounded up, regardless of the value, as the preceding figure shows.

INT function

Like the **EVEN** and **ODD** functions, **INT** is a rounding function. It rounds a number down to the nearest integer.

Uses and applications

INT rounds numbers to the integer value below the number. For instance, 1.99 becomes just 1. In essence, **INT** truncates decimals on any number, leaving the user with just the whole number integer. In that sense, the function is less about rounding than about recovering the integer value in any number. That is where it is useful—when you just want the whole number itself and do not care about the decimals in the value.

The formula

=INT(Value to be rounded)

Figure 5.14 illustrates the use of the INT function:

108	Value	Int	
109	-9.1	-10	=INT(A109)
110	-1.6	-2	=INT(A110)
111	0.3	0	=INT(A111)
112	2.2	2	=INT(A112)
113	3.8	3	=INT(A113)

Figure 5.14: *The INT function*

Errors to avoid

The **INT** function is simple to use—just be aware that it is not really rounding the number. Rather, it is truncating the value and removing the trailing decimal value from a number.

LCM function

The LCM function is Excel's calculator to determine the least common multiple of a set of numbers. While the least common multiples are fairly straightforward to calculate for just two numbers, they can be more difficult to determine for a long list of numbers. The LCM function addresses this issue.

Uses and applications

The least common multiple is a mathematical operation to determine the smallest multiple of two or more numbers. The LCM function determines this value for a set of numbers that the user specifies.

The formula

=LCM(Number 1, Number 2, Number 3, and so on)

All numbers beyond the first one are optional in the formula. However, if you are calculating a least common multiple, you would likely have at least two numbers. *Figure 5.15* illustrates the use of the LCM function:

116	Number 1	Number 2	Number 3	LCM	
117	7	3	12	7	=LCM(A117)
118	7	3	12	21	=LCM(A118:B118)
119	7	3	12	84	=LCM(A119:C119)

Figure 5.15: *The LCM function*

Errors to avoid

The **`LCM`** function is straightforward. The least common multiples themselves are not difficult to calculate, so just make sure you use the intended inputs in the formula.

LN and LOG functions

The LN and LOG functions in Excel calculate the natural log and log of a number, respectively. This is very useful compared to a calculator when dealing with a list of numbers where the log needs to be calculated.

Uses and applications

Logs and natural logs are commonly applied operations in engineering, statistical work, mathematics, and other areas. It is recommended to use these functions instead of a calculator to save time.

The formula

=LN(Number to be referenced)

Or

= LOG(Number to be referenced)

Figure 5.16 illustrates the use of the functions:

121	Number 2	LN	LOG
122	100	4.61	2.00
123	1,000	6.91	3.00
124	1,000,000	13.82	6.00
125		=LN(A124)	=LOG(A124)

Figure 5.16: *The LN and LOG functions*

Errors to avoid

The **LOG** and **LN** functions are quite straightforward—the most likely error would be either using an unintended cell or else trying to calculate the log for an invalid number, such as a negative value (mathematically, logs cannot be calculated on negative values).

MMULT functions

Matrix multiplication is a tedious hassle to do by hand, but the Excel MMULT function makes it quick and easy to do.

Uses and applications

Matrices are commonly used in engineering applications, especially when doing computationally intensive work that can be reduced to a series of matrices. Rather than doing the matrix multiplication by hand (which gets quite difficult for matrices beyond 3×3), Excel can do the work for you using the MMULT function, as long as you have the matrices represented as arrays within Excel cells.

The formula

=MMULT(array 1 values, array 2 values)

Figure 5.17 illustrates the use of the **MMULT** function:

127	Array 1		Array 2	
128	3	4	1	1
129	2	5	1	1
130				
131	7			
132	=MMULT(A128:B129,C128:D129)			

Figure 5.17: *The MMULT function*

Errors to avoid

MMULT is not a complicated function to use. Thus, an error is most likely to occur where either a matrix contains an incorrect value or where an invalid computation (for example, a mismatch in rows and columns) is being attempted.

MOD function

The **MOD** function determines the modulus of a number and a divisor—the remainder when the number is divided by the divisor. This value is simple to calculate by hand or with a calculator in most cases, but it can be tedious with a list of values. This is where the **MOD** function comes in.

Uses and applications

MOD is useful when dealing with a series of values, where you need to find the remaining value after dividing a particular number by a divisor, for instance, looking at remaining dollars after allocating a budget equally to a number of expense categories. As noted, you are most likely to use this either in a dynamic Excel model (such as a budget document) or when dealing with a large list of numbers where the calculation must be done.

The formula

=MOD(number, divisor)

Figure 5.18 illustrates the use of the **MOD** function. Note that the sign of the divisor is the sign the modulus will carry, as the following figure shows:

134	Number	Divisor	MOD	
135	5	2	1	=MOD(A135,B135)
136	5	-2	-1	=MOD(A136,B136)
137	-5	2	1	=MOD(A137,B137)

Figure 5.18: *The MMULT function*

Errors to avoid

MOD is a simple but convenient function for calculations. Be careful not to mix up the divisor or the reference number, and you should not have issues.

POWER function

The **POWER** function is another mathematical operator that saves the user the trouble of doing the calculation manually. In this case, Excel computes the results of one number raised to the power of another number.

Uses and applications

POWER is best used for situations where you have a series of numbers to be raised to power across a list of values; for instance, if you want to do calculations in finance, find the future value of a series of investments made for different time periods and different interest rates.

The formula

=POWER(reference number, power the reference number will be raised to)

Figure 5.19 illustrates the use of the **POWER** function. Note that mathematically, when raising one number to a negative value, the result is the reciprocal of the two positive power operations as a positive number. Hence, 2^–2 is equal to 1/(2 ^2), or ¼.

140	Number	Exponent	POWER	
141	2	3	8	=POWER(A141,B141)
142	2	4	16	=POWER(A142,B142)
143	2	-2	0.25	=POWER(A143,B143)

Figure 5.19: *The POWER function*

Errors to avoid

POWER is an easy enough function to use, so as long as you are careful about which value is the number and which is the exponent, you are unlikely to make an error.

PRODUCT function

The **PRODUCT** function is another mathematical operator that saves the user the trouble of doing the calculation manually. In this case, Excel computes the results of one number multiplied by another number. This is easy to do with a calculator of course, but Excel is excellent for doing this multiplication across lists of numbers very quickly.

Uses and applications

Multiplication is used in numerous applications in every sector of the economy. So, think of this function in Excel as a tool to speed up the process in lieu of a calculator or doing the operation by hand. This cleaner function is also a substitute for simply using the * value in a formula in a cell.

The formula

=PRODUCT(Number 1, Number 2, Number 3, and so on).

The formula will work with two or more numbers—in fact, any number beyond the first one is optional, but in this case, Excel just treats the multiplication being done as the first number times one, so it returns the number itself. That result is not useful in general, so we would likely want at least two numbers in the formula.

Figure 5.20 illustrates the use of the **PRODUCT** function as well as the alternative method using the * sign. Notice that Excel will not allow the use of the more common × sign as an indicator of multiplication.

146	Number 1	Number 2	Number 3	Product	
147	5	10	2	100	=PRODUCT(A147:C147)
148	5	10	2	100	=A148*B148*C148
149	5	10	2	#NAME?	=A149xB149xC149

Figure 5.20: *The PRODUCT function*

Errors to avoid

Like many other math functions, **PRODUCT** is a simple computational function where the most likely error will be in an input value rather than a mistake in applying the formula.

RAND and RANDBETWEEN functions

Excel has a set of functions related to generating a random number or a series of random numbers. **RAND** generates a random number between 0 and 1, whereas **RANDBETWEEN** generates a random number between two values. For those with Excel 365, the new function **RANDARRAY** creates a random array of numbers.

Uses and applications

Random number generation is useful in a variety of situations, such as scenario analysis and Monte Carlo simulation, building test datasets, probability calculations,

and more. Although many mathematical functions could be done (often less efficiently) with a calculator, the random number generation capabilities in Excel are quite unique and not easy to replicate with other tools.

The formula

=RAND()

This function generates a random number between 0 and 1. No arguments in the formula are needed.

=RANDBETWEEN(lower bound value, upper bound value)

This function generates a random integer between the lower and upper bound values.

Figure 5.21 illustrates the two functions. As you will notice, even though the formula is precisely the same for each of the three rows, a different value is generated in each case. In fact, if you use the **RAND()**, you will likely get different values than what is shown here:

152	RAND()	RANDBETWEEN()		
153	0.475	24.000	=RAND()	=RANDBETWEEN(1,100)
154	0.799	63.000	=RAND()	=RANDBETWEEN(1,100)
155	0.406	98.000	=RAND()	=RANDBETWEEN(1,100)

Figure 5.21: *The RAND and RANDBETWEEN functions*

Errors to avoid

RAND() and **RANDBETWEEN** are extremely useful but also extremely quick to cause problems in Excel models. The random values are recalculated each time the sheet is opened or otherwise refreshed, which means that the analysis and numbers in the sheet that relies on these random values can quickly change. It also makes it very hard for someone else to replicate the data. I have often found it easiest to generate random numbers and then copy and hard-paste those numbers over the formulas to avoid these issues.

ROMAN function

The ROMAN function is a fun function that converts a number from the standard Arabic form to a Roman numeral.

Uses and applications

Roman numerals are mainly used in literature and certain scientific references, such as when labeling tables. It is unlikely that we would have a list of numbers to convert to Roman numerals, but Excel would certainly be efficient for that. Instead, the function is most likely to be useful when dealing with a complex Roman numeral, like one derived from a year.

The formula

=ROMAN(number to be converted, form)

Here, form is an optional value that allows you to change between different variations of the Roman numeral system. For the most part, the form value is unnecessary.

The **ROMAN** function is illustrated in *figure 5.22*:

159	Number	Roman
160	1	I
161	2	II
162	3	III
163	4	IV
164	5	V
165	6	VI
166	7	VII
167	8	VIII
168	9	IX
169	10	X
170	2022	MMXXII
171		=ROMAN(A170)

Figure 5.22: *The ROMAN function*

Errors to avoid

It would be difficult to make an error with the **ROMAN** function itself, other than possibly needing a specific form of Roman numeral and using the wrong form option. Make sure to check the output for large numbers where form can matter.

ROUND, ROUNDUP, and ROUNDDOWN functions

Excel has three functions related to rounding numbers; **ROUND**, **ROUNDUP**, and **ROUNDDOWN**. As the names suggest, **ROUND** rounds a number to a given number of decimals based on standard rounding rules, whereas **ROUNDUP** rounds numbers up based on their absolute value. **ROUNDDOWN** rounds numbers down based on their absolute values.

Uses and applications

The Excel rounding functions are useful for taking a broader set of raw data and cleaning it up to the appropriate number of digits. Compared to simply changing the format of the numbers and reducing the number of digits to the desired amount, **ROUND** offers the advantage of actually changing the values, whereas **ROUNDUP** and **ROUNDDOWN** are useful in specific instances where you want the closest value either above or below a given level.

The formula

=ROUND(Number to be rounded, number of digits to round to)

= ROUNDUP(Number to be rounded, number of digits to round to)

=ROUNDDOWN(Number to be rounded, number of digits to round to)

Although the rounding functions themselves are not complicated to implement, the **ROUNDUP** and **ROUNDDOWN** functions can produce some surprising results, as *figure 5.23* shows. In particular, notice that when rounding the number –9.1, **ROUNDUP** produces the value –10, whereas **ROUNDDOWN** produces the value –9. The reason for this is that **ROUNDUP** and **ROUNDDOWN** look at rounding numbers up or down regardless of the sign. Thus, **ROUNDUP** will always increase the magnitude of the numbers regardless of the sign, and **ROUNDDOWN** will do the opposite.

Refer to *figure 5.23* for an illustration of these formulas:

	A	B	C	D	E
173	Value	Round	RoundUp	RoundDown	
174	-9.1	-9	-10	-9	=ROUNDDOWN(A174,0)
175	-1.63	-1.6	-1.7	-1.6	=ROUNDUP(A175,1)
176	0.35	0.4	0.4	0.3	=ROUND(A176,1)
177	2.213	2.21	2.22	2.21	=ROUND(A177,2)
178	3.8000	4	4	3	
179		=ROUND(A178,0)	=ROUNDUP(A178,0)	=ROUNDDOWN(A178,0)	

Figure 5.23: *The ROUND, ROUNDUP, and ROUNDDOWN functions*

Errors to avoid

Be sure to specify the correct number of digits you want numbers rounded to in the formula. Also, be careful when rounding negative numbers that **ROUNDUP** and **ROUNDDOWN** are producing the values that you expected.

SEC and SECH functions

The **SEC** and **SECH** functions are another part of a family of trigonometric functions available in Excel. They give Excel users the secant function and the hyperbolic secant, respectively.

Uses and applications

Similar to the other trigonometric functions, SEC and SECH are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. As before, the advantage of doing the calculation in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, and so if the data are in degrees instead, the function RADIANS should be used.

The formula

=SEC(Reference Cells) and SECH(Reference Cells)

Figure 5.24 shows the use of the **SEC** and **SECH** functions:

182	Value	Secant	Hyperbolic Secant	
183	-3.14	-1.00	0.65	
184	-2.00	-2.40	0.18	
185	0.00	1.00	0.65	
186	2.00	-2.40	0.18	
187	3.14	-1.00	0.65	
188		Input: For SEC and SECH must be smaller than ABS value of 2 ^27		
189		=SEC(A187)	=SECH(B187)	

Figure 5.24: *The SEC and SECH functions*

Errors to avoid

The easiest error to make in the **SEC** function is inputting degrees instead of radians. The range for both **SEC** and **SECH** is any number with an absolute value smaller than 2 to the 27th power—or roughly 134.2 million.

SIGN function

The **SIGN** function is a simple function that returns an indicator variable based on the sign of another number.

Uses and applications

In some circumstances, values with a certain sign may not make sense in a dataset—for instance, negative values of sales or revenues for a business location. Yet, these values can still occur due to errors in the data. Thus, **SIGN** is helpful for finding these abnormal values and cleaning up the dataset.

The formula

=SIGN(Reference number)

Figure 5.25 illustrates that the **SIGN** formula returns a value of 1 for positive number signs, –1 for negative signs, and 0 for values of 0:

192	Value	Sign	
193	103	1	=SIGN(A193)
194	-91	-1	=SIGN(A194)
195	0	0	=SIGN(A195)

Figure 5.25: *The SIGN function*

Errors to avoid

It would be difficult to make an error with the **SIGN** function itself other than possibly mixing up the meaning of the output or referencing the wrong number in the formula itself.

SIN and SINH function

The **SIN** and **SINH** functions are also part of a family of trigonometric functions available in Excel. They give Excel users the sine function and the hyperbolic sine, respectively.

Uses and applications

Similar to the other trigonometric functions, **SIN** and **SINH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. As before, the advantage of doing the calculation

in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, so if the data are in degrees instead, the function **RADIANS** should be used.

The formula

=SIN(Reference Cells) and SINH(Reference Cells)

Figure 5.26 illustrates the use of the **SIN** and **SINH** functions:

198	Value	Sine	Hyperbolic Sine	
199	-3.14	0.00	0.00	
200	-1.57	-1.00	-1.18	
201	0.00	0.00	0.00	
202	0.79	0.71	0.77	
203	3.00	0.14	0.14	
204		Input: For SIN and SINH must be smaller than ABS value of 2 ^27		
205		=SIN(A203)	=SINH(B203)	

Figure 5.26: *The SIN and SINH functions*

Errors to avoid

Again, the easiest error to make in the **SIN** function is inputting degrees instead of radians. The range for both **SIN** and **SINH** is any number with an absolute value smaller than 2 to the 27th power—or roughly 134.2 million.

SQRT function

Excel's square root function SQRT gives users an easy way to take the square root of a number.

Uses and applications

Like some of the other calculation functions, **SQRT** is designed primarily for either dynamic Excel worksheets that contain calculations requiring a square root (for example, determining risk metrics like the Sharpe Ratio in a set of financial data) or for saving time when dealing with large lists of numbers where a square root is needed.

The formula

=SQRT(Reference number)

As *figure 5.27* showing **SQRT** illustrates, Excel cannot deal with taking the square root of a negative number (which mathematically requires the use of imaginary numbers):

208	Value	Square Root	
209	225	15	=SQRT(A209)
210	-9	#NUM!	=SQRT(A210)
211	0	0	=SQRT(A211)

Figure 5.27: *The SQRT function*

Errors to avoid

As noted, Excel will not be able to do the mathematically advanced square roots for negative numbers. Nonetheless, this should not be an issue for most users. Other than that, the function is straightforward in nearly all circumstances.

SUBTOTAL function

The SUBTOTAL function is really a series of functions that allow you to calculate a value based on a list of numbers. For instance, it can calculate the sum, product, or average for a particular list based on the values put into the formula.

Uses and applications

SUBTOTAL is useful for quickly computing different statistics based on a sub-list of values of interest. You could use the individual statistical functions in Excel to accomplish the same thing as **SUBTOTAL** does, but using the function is often more efficient.

The formula

=SUBTOTAL(Numeric value of statistic to be calculated, Cells to reference)

The first argument in the **SUBTOTAL** function is one of 11 different values that correspond to statistics, such as the variance, standard deviation, average, and sum. The second argument in the function is the cells that this statistic should be calculated across.

Figure 5.28 shows the **SUBTOTAL** function is used along with a list of the statistics it can be used to calculate:

213	Value	Subtotal		Type
214	11	9.45	=SUBTOTAL(1,A$214:A$224)	Average
215	14	11.00	=SUBTOTAL(2,A$214:A$224)	Count
216	4	11.00	=SUBTOTAL(3,A$214:A$224)	Count (not empty)
217	20	20.00	=SUBTOTAL(4,A$214:A$224)	Max
218	4	1.00	=SUBTOTAL(5,A$214:A$224)	Min
219	18	1362493440.00	=SUBTOTAL(6,A$214:A$224)	Product
220	8	6.71	=SUBTOTAL(7,A$214:A$224)	Std. Deviation - Sample
221	16	6.40	=SUBTOTAL(8,A$214:A$224)	Std. Deviation - Population
222	6	104.00	=SUBTOTAL(9,A$214:A$224)	Sum
223	1	45.07	=SUBTOTAL(10,A$214:A$224)	Variance - Sample
224	2	40.98	=SUBTOTAL(11,A$214:A$224)	Variance Population

Figure 5.28: *The SUBTOTAL function*

Errors to avoid

The **SUBTOTAL** function is complex because of all of the different statistics it can be used to calculate. Be sure to carefully check that you are calculating the statistic you are interested in and that you have the correct indicator number as the first argument in the function, representing that statistical calculation.

SUMIF, SUMPRODUCT, and SUMSQ functions

Excel has a series of summation functions, most notably the **SUM** function discussed in Chapter 2. While **SUM** is extremely useful, there are some variations that are worth knowing as well, including **SUMIF**, **SUMPRODUCT**, and **SUMSQ**.

Uses and applications

Summing up numbers is a very common task in datasets; **SUM** is, therefore, quite useful. However, we often need to go beyond that, and that is where the other functions come in. SUMIF allows us to sum up particular values that meet certain conditions (like being over a certain size or summing up values for a certain subgroup in a list). **SUMPRODUCT** multiples one list of numbers by another list of numbers and then adds up the result—a very handy operation for all sorts of calculations, such as finding weighted averages. **SUMSQ** squares each number and then adds up all of the values; it is not as commonly used as **SUMPRODUCT**, but it is still a useful timesaver in many calculations.

The formula

=SUMIF(Range to be checked for criteria, Criteria to determine what values should be summed, range of values to be summed up)

= SUMPRODUCT(array 1 of values, array 2 of values)

= SUMSQ(list of values)

Each of the formulas produces a different calculation and statistic, as shown in *figure 5.29*. For **SUMIF**, we can check if a list of numbers meets certain conditions. Those conditions could be related to the numbers themselves or a separate range of related values—and then the function adds up the numbers that meet the criteria. In this case, it is all numbers greater than 5. **SUMPRODUCT** multiplies a list of numbers by another list of numbers—in this case, that means multiplying 11 × 20, 14 × 4, and 4 × 18 and then adding up those resulting values. Finally, **SUMSQ** squares each value and then adds up the group as a whole.

226	Value		
227	11	63	SUMIF
228	14	=SUMIF(A227:A232,">5",A227:A232)	
229	4	348	SUMPRODUCT
230	20	=SUMPRODUCT(A227:A229,A230:A232)	
231	4	1073	SUMSQ
232	18	=SUMSQ(A227:A232)	

Figure 5.29: *The SUMIF, SUMPRODUCT, and SUMSQ functions*

Errors to avoid

These summation functions are very useful, but there are a variety of mistakes that can be made. Probably the most common is either mixing up the sum range and the criteria range in the **SUMIF** function or failing to use the appropriate number of values in the lists being used for **SUMPRODUCT**.

TAN and TANH functions

The **TAN** and **TANH** functions are the final functions that we will discuss in the family of trigonometric functions available in Excel. They give Excel users the tangent function and the hyperbolic tangent, respectively.

Uses and applications

Similar to the other trigonometric functions, **TAN** and **TANH** are specialized functions that are most likely to be used in engineering or math applications where a trigonometric calculator is needed. As before, the advantage of doing the calculation

in Excel versus hard coding the values is that the Excel sheet becomes dynamic. The input angles are in radians, so if the data are in degrees instead, the function **RADIANS** should be used.

The formula

=TAN(Reference Cells) and TANH(Reference Cells)

Figure 5.30 illustrates the use of the **TAN** and **TANH** functions:

235	Value	Tangent	Hyperbolic Tangent
236	-3.14	0.00	0.00
237	-1.57	-1255.77	-1.00
238	0.00	0.00	0.00
239	1.00	1.56	0.91
240	2.00	-2.19	-0.98
241		Input: Any real number	Input: Any real number
242		=TAN(A240)	=TANH(B240)

Figure 5.30: *The TAN and TANH functions*

Errors to avoid

As previously noted with other trigonometric functions, the easiest error to make in the **TAN** function is inputting degrees instead of radians. Be careful about the input values. The range for both **TAN** and **TANH** is any number with an absolute value smaller than 2 to the 27th power—or roughly 134.2 million.

Conclusion

The functions shown in this chapter are all used for various types of mathematical calculations in Excel. Often, there are several ways to do these calculations, either in Excel or with outside tools such as calculators. Nevertheless, these functions are great tools to rely on if you have a large number of calculations to do or if you are trying to build a complex Excel model that will work dynamically.

Points to remember

- Excel has a series of trigonometry functions, including those for sine, cosine, tangent, and more—be sure to use the right function with a valid range and remember that Excel's functions generally use radians as measures of angles.
- Excel's random number functions, including **RAND()** and **RANDBETWEEN()**, can be very useful when building a data set or doing scenario testing where you need random numbers.

- Excel's rounding functions, such as **ROUND**, **ROUNDDOWN**, and **ROUNDUP**, are great tools to use for dealing with large datasets where you need neater numbers with a specific number of significant digits.
- The Excel **SUM** function family is incredibly helpful, starting with the basic **SUM** function itself—but in a lot of cases where you want a conditional sum, **SUMIF** can save you significant time. **SUMPRODUCT** is also very useful for specific scenarios such as weighted averages, looking at probability outcomes, and similar situations.

Multiple choice questions

1. **If you have a series of values representing money collected from individual customers, but you want to determine the amount collected from a subset of those customers who meet a specific criterion, what function would you most likely use?**

   A. SUM

   B. SUMIF

   C. SUMPRODUCT

   D. None of the Above

2. **Column A in a worksheet contains the values of numbers with two decimal places shown, such as 3.57. You want to round all values in Column A down to the nearest whole number. What formula could you use to do this?**

   A. ROUND(Reference Cells)

   B. FLOOR(Reference Cells, 1)

   C. CEILING(Reference Cells, 2)

   D. ROUNDIF(Reference Cells, 1.00)

3. **If you wanted to display the tangent of an angle in a given cell based on another value, what formula could you use?**

   A. =CHOOSE(Reference Cells)

   B. =ANGLE(Tangent Value)

   C. =TAN(Reference Angle in Degrees)

   D. =TAN(Reference Angle in Radians)

Answers

1. **B**
2. **B**
3. **D**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 6

Statistical Functions in Excel

Introduction

Statistical analysis is a foundational cornerstone of modern business, and Excel is a great tool to help with doing that analysis. Although Excel add-ins can add tremendous versatility to the software, even without using them, Excel is still very useful in a variety of situations where statistics are called for. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- NORM.DIST and NORM.INV functions
- PERCENTILE.INC and PERCENTILE.EXC functions
- PHI function
- POISSON.DIST function
- PROB function
- QUARTILE.EXC and QUARTILE.INC functions

- RANK.AVG and RANK.EQ functions
- RSQ function
- SKEW and SKEW.P functions
- STDEVA and STDEVPA functions
- FORECAST and FORECAST.LINEAR functions
- GEOMEAN and HARMEAN functions
- GROWTH function
- LOGNORM.DIST and LOGNORM.INV functions
- MEDIAN function
- MIN function
- NEGBINOM.DIST function
- STEYX function
- T.DIST, T.DIST.2T, and T.DIST.RT functions
- T.INV and T.INV.2T functions
- T.TEST function
- TREND function
- TRIMMEAN function
- VAR.P and VAR.S functions
- Z.TEST function

Objectives

After studying this unit, you will be able to use Statistical functions in Excel for a wide array of tasks.

NORM.DIST and NORM.INV functions

Excel has many statistical functions that enable the calculation of various probability and statistics measures. The most useful statistical distribution functions in Excel are those related to the widely used normal distribution: **`Norm.Dist`** and **`Norm.Inv`**.

Uses and applications

Essentially **`Norm.Dist`** and **`Norm.Inv`** enable the user to calculate the probability of a given value based on the parameters of the normal distribution (**`Norm.Dist`**) or the value in the normal associated with a given probability (**`Norm.Inv`**). The functions also have .S versions for samples versus the population assumption implicit in the basic functions here. These functions can be used for estimating the likelihood of a given outcome in a normal distribution of data or for assessing the outcome with a given likelihood.

The formula

=NORM.DIST(X value, mean of distribution, standard deviation of distribution, 1 or 0 indicator variable for the cumulative density or probability mass function)

= NORM.INV(Probability of value to be calculated, mean of distribution, standard deviation of distribution)

In essence, these formulas allow a user to specify the parameters of the distribution (mean and standard deviation) and then with the **`NORM.DIST`** function calculates the probability that a given X variable will occur (or a value of X or less in the case of a **Cumulative Density Function** [**CDF**]) with the **`NORM.INV`** function, do the reverse, that is, calculate the X variable that will occur with a set probability. *Figure 6.1* illustrates this, calculating the likelihood of X = *3* and then doing the reverse to recover 3 from the associated CDF probability.

	A	B	C
1	X	3	
2	Mean	8	
3	Std. Dev	4	
4			
5	Norm Dist	0.10565	=NORM.DIST(B1,B2,B3,1)
6	Norm Inv	3	=NORM.INV(B5,B2,B3)

Figure 6.1: *The NORM.INV and NORM.DIST functions*

Errors to avoid

Be careful when using **`NORM.DIST`** and **`NORM.INV`** to ensure that your distribution assumptions are valid for the population in the distribution as a whole. If they are not, use **`NORM.S.DIST`** and **`NORM.S.INV`** instead.

PERCENTILE.INC and PERCENTILE.EXC functions

Excel has two functions that examine and report the value at a particular percentile in a distribution—**`PERCENTILE.INC`** and **`PERCENTILE.EXC`**. Both functions report the value at a given percentile—for instance, the 90th percentile on a test score, but the difference between these functions is that the .inc function includes the 90th percentile in the reported value (for example, a test score of 83 out of 100 is the 90th percentile and so anything above that level is above the 90th percentile). In contrast, the **`.exc`** function excludes that 90th percentile value.

Uses and applications

It is quite common to need to know a particular percentile value from a dataset, such as a cut-off or benchmarking measure. For instance, we might want to know the 75th percentile of sales levels for a group of salespeople to know who should get bonuses or the 30th percentile on test scores among students to know who needs additional help.

The formula

=PERCENTILE.INC(List of values to base the percentile on, Percentile to be reported—inclusive of that value)

=PERCENTILE.EXC(List of values to base the percentile on, Percentile to be reported—inclusive of that value)

Figure 6.2 illustrates the **`PERCENTILE`** functions:

9	Person	Income
10	A	1000
11	B	1345
12	C	927
13	D	801
14	E	1176
15	F	1524
16	G	1707
17	30th Percentile	985.4
18		=PERCENTILE.INC(B10:B16,0.3)
19	30th Percentile	956.2
20		=PERCENTILE.EXC(B10:B16,0.3)

Figure 6.2: *The PERCENTILE functions*

Errors to avoid

The most likely error you might make when dealing with the PERCENTILE functions is using the wrong one, that is, using .EXC when you should use .INC or vice versa. The two formulas do produce different results. .EXC tells us what level or value begins the given percentile—what value is needed to just reach that percentile, whereas .INC tells us the value that gets us just beyond that percentile.

PHI function

Despite the name, the PHI function relates to the normal distribution; specifically, it returns the density in a standard normal distribution (for example, mean of 0, standard deviation of 1).

Uses and applications

PHI is useful for figuring out the probability with which a value would occur in a standard normal distribution. While this means a mean of 0 and a standard deviation of 1.0, it is fairly easy to scale that distribution to meet other parameters making the function useful in a wide variety of probability analysis scenarios.

The formula

=PHI(*X*), where *X* is the value from the normal distribution to be used in the probability computation. Remember that with a standard deviation of 1, and a mean of 0, values beyond 3 are quite unlikely to occur, as *figure 6.3* shows:

23	X	Normal Dist Density Value	
24	-3	0.0044	=PHI(A24)
25	-2	0.0540	=PHI(A25)
26	-1	0.2420	=PHI(A26)
27	0	0.3989	=PHI(A27)
28	1.5	0.1295	=PHI(A28)
29	2.5	0.0175	=PHI(A29)
30	4	0.0001	=PHI(A30)

Figure 6.3: *The PHI function*

Errors to avoid

As long as you understand what the standard normal distribution is and can choose the appropriate X, the PHI function is easy to use.

POISSON.DIST function

An alternative probability distribution to the normal is the Poisson distribution. Excel's **POISSON.DIST** function determines cumulative and discrete probability values based on the Poisson distribution.

Uses and applications

The Poisson distribution is an alternative to the normal distribution. The choice of distribution is really just a function of which one fits better with the data, and when the Poisson distribution is the better fit, POISSON.DIST is useful for understanding the likelihood of different outcomes in a dataset.

The formula

=POISSON.DIST(X value, Mean of distribution, Cumulative/Not Cumulative indicator variable).

The function produces either the cumulative probability or the discrete probability of a particular value in the Poisson distribution based on the mean of the distribution. *Figure 6.4* illustrates examples of this:

33	X	9	
34	Mean	8	
35			
36			
37	Poisson Density - Cumulative	0.7166	=POISSON.DIST(B33,B34,1)
38	Poisson Density - Not Cumulative	0.1241	=POISSON.DIST(B33,B34,0)

Figure 6.4: *The POISSON.DIST function*

Errors to avoid

The most likely error when using **POISSON.DIST**, assuming your data fits with a Poisson distribution, of course, is using a cumulative density function when you want noncumulative, or vice-versa.

PROB function

The PROB function is a basic probability calculator—if given a set of values and the probabilities of each, it will calculate the probability of a number between two appearing values.

Uses and applications

The PROB calculator requires a lot of inputs—all possible values and the probabilities associated with them. So it is mainly useful in a dynamic spreadsheet for pulling a probability based on that distribution and then outputting it in an easy-to-read fashion in another part of the sheet. The calculation function itself is not complicated, but the ability to specify the likelihood of outcomes rather than relying on an existing distribution like the Normal is handy.

The formula

=PROB(List of possible values, probability of each of those values, the lower limit for the probability calculation, upper limit for the probability calculation)

Figure 6.5 shows an illustration of the PROB function in action—here, we want to know the likelihood of a value between 4 and 7, given the probability associated with each number. You can envision using this in basic applications, like finding the probability of a given number coming up on the roll of two dice, for instance.

40	Value	Probability
41	1	0.03
42	2	0.06
43	3	0.09
44	4	0.12
45	5	0.2
46	6	0.2
47	7	0.12
48	8	0.09
49	9	0.06
50	10	0.03
51		
52	Probability between 4 and 7	0.64

Figure 6.5 *The PROB function*

Errors to avoid

The PROB function requires a complete set of probabilities for each potential outcome value. If these figures are wrong, the formula calculation will be wrong as well.

QUARTILE.EXC and QUARTILE.INC functions

The quartile functions **QUARTILE.EXC** and **QUARTILE.INC** return the value at the minimum, 25th, 50th, 75th, or maximum percentile. Essentially, they are less versatile versions of the percentile function.

Uses and applications

QUARTILE.EXC and **QUARTILE.INC** are both useful for the same reason that you might use the percentile function—to find a value that exceeds some given threshold. There are no especially good reasons to use them versus the percentile functions other than some minor increase in simplicity.

The formula

=QUARTILE.EXC(Values of Interest, first possible value in the quartile value to be returned)

=QUARTILE.INC(Values of Interest, last possible value in the quartile value to be returned)

The second argument in the function is a number from 0 to 4 that tells Excel whether the function should return the minimum (0), 25th percentile (1), 50th percentile (2), 75th percentile (3), or maximum from the data points.

Like the percentile functions, .INC and .EXC represent the ceiling and floor of the given percentile. .INC indicates that the value returned includes the percentile in question, whereas .EXC means that it excludes that value. As *figure 6.6* shows, you can replicate the output from the quartile functions with the percentile functions by just choosing 0.25, 0.5, 0.75, and so on as the percentile to be returned.

55	Person	Income	
56	A	1,000.00	
57	B	1,345.00	
58	C	927.00	
59	D	801.00	
60	E	1,176.00	
61	F	1,524.00	
62	G	1,707.00	
63	25th Percentile	963.50	=PERCENTILE.INC(B56:B62,0.25)
64	25th Percentile	963.50	=QUARTILE.INC(B56:B62,1)
65	25th Percentile	927.00	=QUARTILE.EXC(B56:B62,1)
66	75th Percentile	1,434.50	=QUARTILE.INC(B56:B62,3)

Figure 6.6: *The NORM.INV and NORM.DIST functions*

Errors to avoid

As with the percentile functions, one likely mistake when using **QUARTILE** functions is selecting **.EXC** when **.INC** is needed, or vice versa. The other mistake that is fairly likely is picking the wrong value for the quartile to be returned, especially not including a value (which will default the selection to 0).

RANK.AVG and RANK.EQ functions

Excel's rank functions **RANK.AVG** and **RANK.EQ** are very useful when dealing with a large dataset and trying to rank certain categories in order.

Uses and applications

The RANK functions are extremely helpful when dealing with panel datasets where we have a series of categories with values that occur over time. For instance, we might be interested in ranking a series of stores based on their average daily sales. The rank functions let us do that without having to resort to the values or otherwise alter the underlying dataset.

The formula

=RANK.AVG(Value to be ranked, comparison dataset)

= RANK.EQ(Value to be ranked, comparison dataset)

Initially, **RANK.AVG** and **RANK.EQ** might seem the same—most of the time, they produce the same values. However, if there are duplicate values in a dataset to be ranked, as in *figure 6.7*, then **RANK.AVG** will give the average rank across all these equal values, whereas **RANK.EQ** will give all of the equal values the same top rank.

In the following example, where 1,000 is used as an income twice, under **RANK.EQ** results in a value of 5 (and there is no rank 6 because there are two rank 5 values), while **RANK.AVG** gives a value of 5.5 for both ranks—the average of 5 and 6 for the two values.

69	Person	Income	Rank.Avg	Rank.Eq
70	A	1,000.00	5.5	5
71	B	1,345.00	3	3
72	C	1,000.00	5.5	5
73	D	801.00	7	7
74	E	1,176.00	4	4
75	F	1,524.00	2	2
76	G	1,707.00	1	1
77			=RANK.AVG(B76,B$70:B$76)	=RANK.EQ(B76,B$70:B$76)

Figure 6.7: *The NORM.INV and NORM.DIST functions*

Errors to avoid

The rank functions are easy to use; just be sure to anchor your cells if dragging the formula to apply it to a series of values across a dataset to be ranked.

RSQ function

The RSQ function gives the squared correlation between two sets of data points. Specifically, it gives a type of correlation called the Pearson product-moment and then squares the value. This is similar to the more widely known **CORREL** function—but **CORREL** is the square root of RSQ.

Uses and applications

The squared correlation is more generally called R-squared, and it is an important statistical measure of how well a regression line fits a set of data. Essentially, you can use RSQ to find this value manually.

The formula

=RSQ(Y variables, X variables)

Figure 6.8 illustrates the RSQ function:

79	Y Variable	X Variable
80	4	11
81	7	14
82	5	17
83	8	12
84	3	9
85	2	6
86	5	15
87		
88	0.3728	=RSQ(A80:A86,B80:B86)

Figure 6.8: *The RSQ function*

Errors to avoid

The R-squared value is easy to compute with the RSQ function—just make sure you understand how to interpret the output correctly.

SKEW and SKEW.S functions

Skewness is a statistical measure of the shape of the distribution of data. Excel has two primary functions for measuring Skew; **`SKEW.P`** and **`SKEW`**, which measure the skew for a complete population of data and for a representative sample, respectively.

Uses and applications

Skewness is a number that tells us whether a distribution of data is "tilted" toward the positive or negative side, and it is often a result of the type of data being examined. For instance, in a large sample of income data, there will always be a positive skew because no one can have a negative income, and a few people will have very large positive incomes. Hence, income is bounded by 0 on the downside (a person who has no income) but has no upper bound. In these circumstances, we need to make certain statistical adjustments to the data or at least acknowledge the skew for reporting purposes. The Excel functions give us a measure of how large that skew is.

The formula

=SKEW(Reference Cells)

= SKEW.P(Reference Cells)

The difference between **`SKEW`** and **`SKEW.P`** is simply what the data represent. If the data are a complete dataset of all observations we care about (for instance, the grade point average of all students in a graduating cohort of university students), then we can use SKEW.P. If the data are supposed to represent a sample from a larger population where we do not have complete information, then we can use **`SKEW`**.

Figure 6.9 illustrates both formulas in use:

	A	B	C
93	Person	Income	
94	A	1000	
95	B	1345	
96	C	927	
97	D	801	
98	E	1176	
99	F	1524	
100	G	1707	
101	Skew - Population	0.263583235	=SKEW.P(B94:B100)
102	Skew - Sample	0.341642919	=SKEW(B94:B100)

Figure 6.9: *The SKEW functions*

Errors to avoid

SKEW and **SKEW.P** are simple to implement, but just be certain to use the correct formula for your application.

STDEVA and STDEVPA functions

There are a number of standard deviation functions in Excel, but two of the most useful ones are the **STDEVA** and **STDEVPA**. Unlike other standard deviation functions, the functions are able to deal with text data and missing values in a dataset.

Uses and applications

Standard deviation in a dataset is one of the most widely used and important statistical measures. It captures the level of variation in a dataset in a convenient number, but the normal standard deviation functions such as **STDEV.P** simply ignore any text values or missing values in the dataset. If you want to include those values in the calculation as zeros, you can do that using **STDEVA** and **STDEVPA**.

The formula

=STDEVA(Reference Cells)

=STDEVPA(Reference Cells)

The **STDEVA** function calculates the standard deviation for a sample of data and includes all text values or missing values with a zero in the calculation. The **STDEVPA** function fulfills a similar role but does the calculation for a full population of all available data points. Both results differ markedly from the results from the basic **STDEV.S** and **STDEV.P** functions, which ignore text values. *Figure 6.10* illustrates this:

	A	B	C
104			
105	Person	Income	
106	A	1000	
107	B	1345	
108	C	927	
109	D	801	
110	E	1176	
111	F	1524	
112	G	1707	
113	H	No data	
114	Standard Deviation	526.5190269	=STDEVA(B106:B113)
115	Std Deviation - Population	492.5134516	=STDEVPA(B106:B113)
116	Std Deviation - Basic form	306.2309386	=STDEV.P(B106:B113)

Figure 6.10: *The STDEVA and STDEVPA functions*

Errors to avoid

STDEVA and **STDEVPA** are easy functions to use; the most common error here is using one of these functions when you should be using **STDEV.P** or **STDEV.S**. Be sure to think carefully about whether it is appropriate to include logical and text values as zeros in the dataset. Note that if the value is a logical value equal to true, then it is included in the dataset as a 1.0, not 0.

FORECAST and FORECAST.LINEAR functions

FORECAST and **FORECAST.LINEAR** are handy functions for using Excel to quickly make projections. The functions are built around regressions. Regressions can be done manually, but **FORECAST** and **FORECAST.LINEAR** take the process from being a multistep affair to being quick and efficient.

Uses and applications

FORECAST and **FORECAST.LINEAR** both rely on regression-based approaches to forecasting. This is a very useful tool for many business applications, but to make the forecasts effective, it is important to be sure that you have a large and robust set of X variables (the driving factors behind the value you are trying to predict) and Y variables (the outcome variable resulting from the driving factors). Predicting sales, inventory, and consumer trends are all good examples of where this type of forecasting often makes sense.

The formula

=FORECAST(Value of X the outcome variable Y should be based on, Known Y variables, Known X variables)

=FORECAST.LINEAR(Value of X the outcome variable Y should be based on, Known Y variables, Known X variables)

Both functions fulfill the same purpose; **FORECAST.LINEAR** is part of an expansion to the basic forecasting toolset that started to allow other methods of forecasting.

Figure 6.11 illustrates both formulas in use:

119	Person	Income	Age
120	A	1000	29
121	B	1345	47
122	C	927	22
123	D	801	26
124	E	1176	31
125	F	1524	54
126	G	1707	46
127	Forecast	1177.353033	35
128		=FORECAST(C127,B120:B126,C120:C126)	
129	Linear Forecast	1344.323171	42
130		=FORECAST.LINEAR(C129,B120:B126,C120:C126)	

Figure 6.11: *The NORM.INV and NORM.DIST functions*

Errors to avoid

The most common error in using forecast is likely mixing up the X and Y variables. Remember, X is the variable that drives Y, and Y is the variable you want to predict.

GEOMEAN and HARMEAN functions

We have already discussed Excel's **AVERAGE** function used for calculating the arithmetic mean of a set of numbers; however, in certain circumstances, the arithmetic mean is not the appropriate way to do an average. In these cases, Excel has **GEOMEAN** and **HARMEAN** to allow the calculation of the geometric mean and harmonic mean, respectively.

Uses and applications

When dealing with growth rates, we cannot use an arithmetic mean to find the average growth rate. For instance, for an investment portfolio, if we have annual returns data for a given number of years, the arithmetic mean will give us an incorrect answer when trying to find the average rate of growth for the period. In this case, a geometric mean is needed. Similarly, a harmonic mean is needed when dealing with ratios, where small changes in the numerator or denominator can have very large impacts on the overall value (for example, the difference between an asset base of $100 and an asset base of $2,000 against profits of $500 leads to a profit/asset ratio of 5 versus 0.25).

The formula

=GEOMEAN(Reference cells)

= HARMEAN(Reference cells)

Harmonic means and geometric means are easy to calculate but need to be used in the correct circumstances. An illustration of their use is shown in *figure 6.12*:

133	Year	Price/Earnings Ratio	Investment Growth Rate
134	2022	9	5%
135	2023	12	8%
136	2024	15	11%
137	2025	8	7%
138	2026	22	19%
139	2027	13	12%
140	2028	15	4%
141		12.1706238	0.083410513
142		=HARMEAN(B134:B140)	=GEOMEAN(C134:C140)

Figure 6.12: *The HARMEAN and GEOMEAN functions*

Errors to avoid

Be careful to use the correct type of average for your data. Use **GEOMEAN** with growth rates, **HARMEAN** with ratios, and **AVERAGE** with other types of numeric data.

GROWTH function

The **GROWTH** function is an easy way to do forecasts in Excel for exponentially growing values.

Uses and applications

For data where growth is exponential, such as population data or asset values, linear regression-based forecasting may struggle. With the **GROWTH** function, Excel allows users to apply an exponential growth function to a set of data and predict values accordingly.

The formula

=GROWTH(Y Values to base prediction on, X values driving the Y value, X value corresponding to value to be predicted)

Figure 6.13 illustrates using the GROWTH function to predict a value based on previous years' values and the year for which the value should be predicted.

	A	B	C
142			
143	Year	Value	
144	2022	100	
145	2023	130	
146	2024	170	
147	2025	250	
148	2026	320	
149	2027	410	
150	2028	540	
151	2029	734.5678033	
152		=GROWTH(B144:B150,A144:A150,A151)	

Figure 6.13: *The GROWTH function*

Errors to avoid

GROWTH cannot account for multiple X variable driving factors, so be sure to use it in an appropriate setting. In addition, while the formula will allow you to omit the (optional) X variables and the new X value, doing so will often result in strange and nonsensical forecasts.

LOGNORM.DIST and LOGNORM.INV functions

LOGNORM.DIST and **LOGNORM.INV** are additional statistical functions related to doing calculations on statistical distribution functions; in this case the log-normal distribution.

Uses and applications

Like the normal distribution functions, **LogNorm.Dist** and **LogNorm.Inv** enable the user to calculate the probability of a given value based on the parameters of a distribution. However, in this case, the log-normal distribution (**Norm.Dist**) and the value in the log normal are associated with a given probability (**LogNorm.Inv**).

The formula

=LOGNORM.DIST(X value, mean of distribution, standard deviation of distribution, 1 or 0 indicator variable for the cumulative density or probability mass function)

= LOGNORM.INV(Probability of value to be calculated, mean of distribution, standard deviation of distribution)

In essence, these formulas allow a user to specify the parameters of the distribution (mean and standard deviation) and then with the **`LOGNORM.DIST`** function, calculate the probability that a given X variable will occur (or a value of X or less in the case of a **cumulative density function** [CDF]), and with the **`LOGNORM.INV`** function, do the reverse, that is, calculate the X variable that will occur with a set probability.

Figure 6.14 illustrates this, calculating the likelihood of X = 3 and then doing the reverse to recover 3 from the associated CDF probability.

	A	B	C	D
154				
155	X	3		
156	Mean	8		
157	Std. Dev	4		
158				
159	Norm Dist	0.042232485	=LOGNORM.DIST(B155,B156,B157,1)	
160	Norm Inv	3	=LOGNORM.INV(B159,B156,B157)	

Figure 6.14: *The LOGNORM.INV and LOGNORM.DIST functions*

Errors to avoid

As with other distribution functions, be careful with **`LOGNORM.DIST`** and **`LOGNORM.INV`** to be sure that your distribution assumptions are valid for the population in the distribution as a whole.

MEDIAN function

Although Excel's **`PERCENTILE`** functions will allow you to find any percentile in a series of data, in many cases, it is more intuitive to use a function tied to a specific percentile of note, and the **`MEDIAN`** function is an example of this.

Uses and applications

`MEDIAN` finds the 50th percentile of a given set of numbers. Using the median avoids some of the potential issues with outliers that can impact simple averages, and Excel makes it easy to calculate this value.

The formula

=MEDIAN(Reference cells)

Figure 6.15 illustrates the use of the **MEDIAN** function:

	A	B	C
163	Person	Income	
164	A	1000	
165	B	1345	
166	C	927	
167	D	801	
168	E	1176	
169	F	1524	
170	G	1707	
171	Median	1176	=MEDIAN(B164:B170)

Figure 6.15: *The MEDIAN function*

Errors to avoid

The **MEDIAN** function is quite simple to use, so you are unlikely to make any errors when using it other than potentially pointing it to the wrong set of reference cells.

MIN function

In many cases, the things that Excel users need to do with a dataset are pretty simple. For instance, finding the biggest value or smallest value, and the latter is what the MIN function does.

Uses and applications

While Excel models can be very complex, in many cases, they are built on simple tools like **MIN** and **MAX**. The MIN function lets the user find the lowest value in a set of numbers, which is useful for a range of activities, from finding outliers to building more complex functions.

The formula

=MIN(Reference Cells)

Figure 6.16 illustrates the use of the **MIN** function:

	A	B	C
173			
174	Person	Income	
175	A	1000	
176	B	1345	
177	C	927	
178	D	801	
179	E	1176	
180	F	1524	
181	G	1707	
182	Min	801	=MIN(B175:B181)

Figure 6.16: *The MIN function*

Errors to avoid

The **MIN** function is simple to understand and simple to use. Double-check the reference cells, and you should avoid any issues.

NEGBINOM.DIST function

The **NEGBINOM.DIST** function produces the likelihood of a given outcome under a negative binomial distribution.

Uses and applications

A negative binomial distribution function predicts the likelihood of a given outcome based on the number of successes, the number of failures, and the probability of success. For instance, the negative binomial could be used to examine the likelihood of at least half the students in a class passing an exam, given the probability of any single student passing (which could be drawn from past exam results). Essentially NEGBINOM.DIST is a calculator for the chances of a given event happening.

The formula

=NEGBINOM.DIST(Number of failures, number of successes, probability of success, cumulative or discrete indicator)

The function requires the number of successes and failures as inputs, along with the probability of a failure. The final value to be chosen is a cumulative (1) or discrete (0) indicator variable. In essence, this variable for the cumulative option produces the probability of at least X number of successes in a distribution, whereas discrete produces the probability of exactly that outcome.

Figure 6.17 illustrates the use of **NEGBINOM.DIST** function:

185	Number of Failures	8	
186	Number of Successes	3	
187	Probability	0.3	
188			
189	Norm Dist	0.687259546	=NEGBINOM.DIST(B185,B186,B187,1)

Figure 6.17: *The NEGBINOM.DIST function*

Errors to avoid

Be careful to make sure that the formula selects the correct variables. For instance, if the number of successes is used in the formula where the number of failures is supposed to be used, the calculation will be incorrect.

STEYX function

Excel may not have the regression capabilities of some specialized software packages, but it still has all the capabilities needed to do basic regressions for most users. The STEYX function fits right into the category of function.

Uses and applications

In regression analysis, standard errors are important as measures of accuracy and confidence in the point estimates provided by the regression coefficients. STEYX provides this standard error; that is, you can use the function to produce a standard error based on a set of X and Y variables. The standard error is also an input variable used in the calculation of other outputs for regression analysis.

The formula

=STEYX(Set of Y variables, Set of X variables)

Excel has multiple regression capabilities, but it is limited in its ability to handle multiple regression in formulas. Instead, simple linear regression can be done with formulas, and **STEYX** is a key part of that.

Figure 6.18 illustrates the use of the **STEYX** function:

192	Person	Income	Age	
193	A	1000	29	
194	B	1345	47	
195	C	927	22	
196	D	801	26	
197	E	1176	31	
198	F	1524	54	
199	G	1707	46	
200	Forecast	1177.353033	35	
201	Standard Error	165.2725504	=STEYX(B193:B199,C193:C199)	

Figure 6.18: *The STEYX function*

Errors to avoid

STEYX is an easy function to use. Just be sure not to mix up X and Y variables in the formula.

T.DIST, T.DIST.2T, and T.DIST.RT functions

Excel has three functions related to the T-distribution, or Student's T-Distribution: T.DIST, T.DIST.2T, and T.DIST.RT. All of them calculate distributional results from the T-distribution given an X and a specified degree of freedom.

Uses and applications

Like the other distribution functions for Normal, Log Normal, and other probability distributions, the T-distribution functions are mainly useful when trying to calculate the value that results from a given distributional assumption.

The formula

=T.DIST(X-value, degrees of freedom, cumulative indicator)

=T.DIST.2T(X-value, degrees of freedom)

=T.DIST.RT(X-value, degrees of freedom)

All three functions serve a similar purpose but with slight variations. The basic **T.DIST** function can be used to calculate either a probability distribution function or a cumulative density function for a given X value. The **T.DIST.2T** function calculates the two-tailed T-distribution probability function, whereas the **T.DIST.RT** function

calculates the right-tailed t-distribution probability function. *Figure 6.19* illustrates the use of the three functions:

205	X	2	
206	Degrees of Freedom	30	
207			
208	T- Distribution	0.972687478	=T.DIST(B205,B206,1)
209	2 Tailed T-Distribution	0.054625045	=T.DIST.2T(B205,B206)
210	Right Tailed T-Dist.	0.027312522	=T.DIST.RT(B205,B206)

Figure 6.19: *The T.DIST functions*

Errors to avoid

A common error in using many of the distributional functions is using a cumulative density function when a probability density function should be used instead of vice-versa. Be sure you understand which type you are looking for in your specific application.

T.INV and T.INV.2T functions

Excel has many statistical functions that enable the calculation of various probability and statistics measures. The **`T.INV`** and **`T.INV.2T`** functions enable Excel users to find the resulting T distribution value, given a probability and degrees of freedom.

Uses and applications

In some statistical applications, we may wish to know the value that would be achieved with a specified probability. Inverse functions enable that calculation. In the case of the T-distribution, Excel has a one-tailed and two-tailed versions: **`T.INV`** and **`T.INV.2T`**.

The formula

=T.INV(probability, degrees of freedom)

=T.INV.2T(probability, degrees of freedom)

The two functions have the same inputs—the degrees of freedom and the probability in the distribution of the value to be determined. The **`T.INV.2T`** performs the calculation for a two-tailed distribution instead of a one-tailed (**`T.INV`**), but both calculate the X variable that will occur with a set probability.

Figure 6.20 illustrates this, calculating the value that will occur with a given probability—in each case, the recovered value is 2:

214	Degrees of Freedom	30	
215	T- Distribution	0.972687478	
216	2 Tailed T-Distribution	0.054625045	
217			
218	T-inverse	2	=T.INV(B215,B214)
219	2 Tail T-inverse	2	=T.INV.2T(B216,B214)

Figure 6.20: *The NORM.INV and NORM.DIST functions*

Errors to avoid

Be sure that you know whether you want a distributional function that will calculate a probability (for example, **`T.DIST`**) or the inverse function that calculates the value resulting from a given probability (for example, **`T.INV`**).

T.TEST function

The student's *t*-test is a common tool in basic statistics for evaluating if there is a statistical difference between the two mean values of two groups. The T.TEST function in Excel performs this test.

Uses and applications

T-tests are common for evaluating means between different sets of values or between a value and the mean of a set of data. The calculation can be done by hand, but it can be a bit tedious, so **`T.TEST`** makes like much easier when doing the calculation.

The formula

=T.TEST(first set of values, second set of values, 1 or 2 tail test indicator, type of test to be done)

The **`T.TEST`** function has a variable that determines the type of test to be performed—if the last argument is 1, a paired test will be performed. If the last argument is 2, a two-sample equal variance test is done, and if the value is 3, a two-sample unequal variance test is done.

Figure 6.21 illustrates these tests:

222	Variable A	Variable B	
223	10	29	
224	134	47	
225	92	22	
226	8	26	
227	11	31	
228	15	54	
229	17	46	
230			
231	T-Test - 1 Tail Paired	0.410671	=T.TEST(A$223:A$229,B$223:B$229,1,1)
232	T-Test, 2 Tailed Paired	0.821342	=T.TEST(A$223:A$229,B$223:B$229,2,1)
233	T-Test, 2 Tailed Equal Var	0.820763	=T.TEST(A$223:A$229,B$223:B$229,2,2)
234	T-Test, 2 Tailed Unequal Var	0.823767	=T.TEST(A$223:A$229,B$223:B$229,2,3)

Figure 6.21: *The T.TEST function*

Errors to avoid

The **T.TEST** function can be complex. Be sure that you understand the options in the function for a one-tailed and two-tailed test and the type of test to be done.

TREND function

Excel has a number of basic forecasting functions, and **TREND** is one of those. While similar in general to **FORECAST**, **TREND** offers a distinct flexibility that sets it apart.

Uses and applications

Forecasting is important in a variety of business situations, and TREND is a simple way to do a forecast given even a small amount of data. While the forecast method that TREND uses is basic, it is a great starting point for more complex forecast methods.

The Formula

=TREND(Y variables, X variables, x-variable to base predicted y variable off of, indicator value for b = 1)

Like **FORECAST**, **TREND** works based on the concept of simple linear regression. It offers an easy way to do a forecast if you have a set of values to base your prediction on. For instance, in *figure 6.22*, income is predicted based on a linear regression of age. The distinction of **TREND** compared with **FORECAST** is that **TREND** enables the user to choose whether they want the baseline slope intercept "b" value to be equal

to 1.0 or unconstrained and driven by the data. The appropriate choice will depend on the circumstances and the data to be analyzed.

237	Person	Income	Age	
238	A	1000	29	
239	B	1345	47	
240	C	927	22	
241	D	801	26	
242	E	1176	31	
243	F	1524	54	
244	G	1707	46	
245	Trend, b = 1	1134.452122	35	=TREND(B238:B244,C238:C244,C245,0)
246	Trend, b <> 1	1177.353033	35	=TREND(B238:B244,C238:C244,C246,1)

Figure 6.22: *The TREND function*

Errors to avoid

Be careful to select the correct data for Y (the value to be predicted) versus X, the value that the prediction is based upon.

TRIMMEAN function

Sometimes when analyzing data, it is helpful to exclude outliers, the extreme top and bottom variables in the dataset. This is particularly true when dealing with large datasets with lots of observations. Excel makes it easy to remove these outliers using the TRIMMEAN function.

Uses and applications

TRIMMEAN enables the user to remove a set percentage of the top and bottom outlier variables in the data. This can help with getting a cleaner interpretation of the average of datasets that have a lot of noise or missing values in them.

The formula

= TRIMMEAN(reference variables, percentage of data points to be removed)

TRIMMEAN takes the specified percentage of data points and removes half of that percentage from the top outliers in the data and half from the bottom outliers in the

data. *Figure 6.23* illustrates the result looking at **TRIMMEAN** with a 30% reduction in data points versus the standard **AVERAGE** function.

249	Person	Income		
250	A	1000		
251	B	1345		
252	C	927		
253	D	801		
254	E	1176		
255	F	1524		
256	G	1707		
257	Mean	1211.43	=AVERAGE(B250:B256)	
258	Trimmed Mean	1194.40	=TRIMMEAN(B250:B256,0.3)	

Figure 6.23: *The TRIMMEAN function*

Errors to avoid

Be careful deciding what percentage of data points to remove—**TRIMMEAN** is not necessarily better than **AVERAGE** in some cases; it really depends on the data set and how many data points should be removed to stop them from skewing the output.

VAR.P and VAR.S functions

Like standard deviation, the variance is an extremely common statistic to need to calculate, and Excel has two functions to do it: **VAR.S** and **VAR.P**.

Uses and applications

Variances are widely used as measures of dispersion within a set of data. They are the square of standard deviation and given a numeric measure for how widely spread out a data set is. Variance is commonly used in finance, and many statistical calculations, but a key issue that must be resolved is whether the user is looking for the variance of the population (because they have a complete set of all data of interest) or the variance of a representative sample of the larger data set. That is the difference between **VAR.P**—population—and **VAR.S**—sample.

The formula

=VAR.P(reference data points)

=VAR.S(reference data points)

Figure 6.24 illustrates the use of the sample variance calculation, **VAR.S** and the population variance calculated with **VAR.P**. Notice that the values produced are different—the variance calculation that should be used varies depending on the circumstances and whether the user has the complete set of all data in the population. Please refer to the following figure:

262	Person	Income	
263	A	1000	
264	B	1345	
265	C	927	
266	D	801	
267	E	1176	
268	F	1524	
269	G	1707	
270	Variance - Sample	109,406.95	=VAR.S(B263:B269)
271	Variance - Population	93,777.39	=VAR.P(B263:B269)

Figure 6.24: *The VAR.P and VAR.S functions*

Errors to avoid

The biggest error likely to be made when dealing with the variance functions is using **VAR.P** when **VAR.S** should be used, or vice versa.

Z.TEST function

The Z-test is a very common statistical test that examines whether a particular value is greater than the hypothesized mean of a data set. Excel offers a **Z.TEST** function for use in computing Z-statistics.

Uses and applications

The Z-test is typically used to determine if the means between two data sets are equal—but Excel refines this to examine the probability of whether the value to be tested is greater than the mean of a data sample.

The formula

=Z.TEST(array of values in data set, value to be test against the mean of the array, (optional – sigma value))

The **Z.TEST** function produces the probability that the value to be tested is greater than the mean of a data set represented by a given set of data—in *Figure 6.25*, variable A values. Although we know with certainty what is the mean of variable A's values,

we do not know what the true mean of the population for variable A is—and that is what **Z.TEST** is checking the value against.

274	Variable A	Value to be tested	
275	10	31	
276	134		
277	92		
278	8		
279	11		
280	15		
281	17		
282			
283	Z-test	0.301060	=Z.TEST(A$223:A$229,B275)

Figure 6.25: *The Z.TEST function*

Errors to avoid

The sigma argument in the function is optional. Be sure to decide if it is appropriate for a given use case before using the function.

Conclusion

The functions shown in this chapter are statistical functions. They are useful in many circumstances when you need to perform statistical tests or otherwise examine large datasets. Although Excel does not have the statistical capabilities of some other software packages, it is more than enough for many business users, especially if you understand the functions that have been covered here.

Points to remember

- Excel has a series of statistical functions aimed at calculations around probability distributions, including those for normal distributions, T distributions, Poisson distributions, and more. Be sure to use the right function with a valid set of inputs.
- Excel's forecasting tools, such as **TREND** and **FORECAST**, are great options for quick and easy forecasts if you have a set of data you want to examine.
- Excel's rounding functions for ranking—**RANK.AVG** and **RANK.EQ** are very useful for evaluating performance in a dataset, such as for employees, business locations, people in a group, and so on.
- Although we think about the arithmetic mean when we think of an average, Excel has two other means that may be a better choice to use in some circumstances: **HARMEAN** and **GEOMEAN**.

Multiple choice questions

1. **The MEDIAN function is equivalent to using the PERCENTILE function with what percentile specified?**
   A. 25
   B. 50
   C. 75
   D. Not enough information

2. **What function could be used to recover the X value that occurs in a Normal distribution with a given probability?**
   A. NORM.DIST
   B. NORM.INV
   C. T.INV
   D. Not enough information

3. **What function could be used to evaluate if two sets of numbers have the same mean based on a statistical significance test?**
   A. PROB
   B. RSQ
   C. TREND
   D. T.TEST

Answers

1. **B**
2. **B**
3. **D**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 7

Text Functions in Excel

Introduction

Excel can help us clean up and organize datasets effectively, which is especially important when dealing with mixed or semi-structured datasets. However, Excel can only help if we can use text functions proficiently. In this chapter, we will cover the most popular and useful formulas you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- CLEAN function
- CODE function
- CONCATENATE function
- DOLLAR function
- EXACT function
- FIND function

- FIXED function
- LEFT and RIGHT functions
- LEN function
- LOWER and UPPER functions
- MID functions
- NUMBERVALUE function
- PROPER function
- REPLACE functions
- SEARCH functions
- SUBSTITUTE function
- T function
- TEXT function
- TRIM function
- UNICHAR function
- UNICODE function
- VALUE function

Objectives

After studying this unit, the reader will be able to use a variety of text functions in Excel.

CLEAN function

The **CLEAN** function is primarily useful when you have pulled in large amounts of text data and want to remove any non-printable characters.

Uses and applications

Oftentimes, data imported from other sources may contain non-printable characters or low-level computer code that is essentially just gibberish to a human reader. To remove these characters and get clean text, Excel has the **CLEAN** function.

The formula

=CLEAN(Text to be cleaned, or reference cell containing the text)

Figure 7.1 illustrates the use of the **CLEAN** function:

	A	B	C	D
1	Print This		Print This	
2	=CLEAN(C1)		=CHAR(9)&"Print This"&CHAR(10)	
3				

Figure 7.1: *The CLEAN function*

Errors to avoid

The **CLEAN** function is easy to use—just make sure you are referring to the text that you want to be cleaned, and you should have no trouble.

CODE function

The **CODE** function returns a number that represents the first character in a text string.

Uses and applications

CODE is useful for sorting, matching, classifying, and otherwise starting to turn text data into numeric code. The values after the first character in the string do not impact the returned character—**CODE** is just assigning a number to whatever the first character in the referenced string is.

The formula

=CODE(Text to be "coded", or reference cell containing the text)

Figure 7.2 illustrates the use of the **CODE** function:

	A	B	C
6	65		Alphabet
7	69		Excel
8	69		Elephant
9	82		Read This
10	=CODE(C9)		

Figure 7.2: *The CODE function*

Errors to avoid

The trickiest part of the **CODE** function is recognizing that it is not encoding the entire text string—the first character is assigned a number, but the other characters in the string after the first have no impact on the returned value. This is illustrated in the preceding figure, where both "Elephant" and "Excel" return the same value of "69". However, upper case and lower case letters produce different values—a lower case "elephant" produces a value of 101 instead of 69, for instance.

CONCATENATE function

When you want to combine two text strings together into a single value—for instance, a first and last name in separate cells, Excel's **CONCATENATE** function is your best option.

Uses and applications

There are many instances in datasets where you may want to combine two string variables together, such as creating full names, unique identifiers (for example, Worker 1 from Department A), or combining elements in an address. **CONCATENATE** enables the user to do this quickly and effectively.

The formula

=CONCATENATE(Cell 1 to be combined, Cell 2 to be combined, and so on.)

Figure 7.3 illustrates the use of the **CONCATENATE** function:

14	Street,City,Country		Street
15	=CONCATENATE(C14,",",C15,",",C16)		City
16			Country

Figure 7.3: *The CONCATENATE function*

Errors to avoid

Like many other text functions, **CONCATENATE** is simple to use but powerful—just be sure to put any direct text that you want in the formula in the " " marks.

DOLLAR function

The **DOLLAR** function is a simple function to change a standard number value to a dollar text value.

Uses and applications

When dealing with large amounts of data, sometimes currency figures end up being imported into Excel as numbers rather than currency values. To deal with this, the **DOLLAR** figure converts any number to a text string in dollar currency terms.

The formula

=DOLLAR(number to be expressed as a dollar amount, optional number of decimal places to display)

Figure 7.4 illustrates the use of the **DOLLAR** function:

18	$1.00
19	=DOLLAR(1)

Figure 7.4: *The DOLLAR function*

Errors to avoid

The **DOLLAR** function is quite straightforward—the only place an error is likely to occur is if you are referencing another cell—for instance, taking the value in cell B24 and turning it into a dollar value. In this case, just make sure the formula references the intended cell. Other than that, errors are unlikely.

EXACT function

The **EXACT** function enables the Excel user to check two values against one another—if the values are the same, then **EXACT** will return the value TRUE, but if they are different, **EXACT** will return the value **FALSE**.

Uses and applications

In many cases, when using large amounts of data, we may want to match two datasets together based on a common identifier present in both datasets, such as a person's last name. However, because of formatting or data errors, sometimes the common identifiers may not match. That is where **EXACT** is useful.

The formula

=EXACT(Text value 1, Text value 2)

Figure 7.5 illustrates the use of the **EXACT** function:

22	TRUE	SAM	SAM
23	FALSE	Sam	SAM
24	=EXACT(B22,C22)		

Figure 7.5: *The EXACT function*

Errors to avoid

The **EXACT** function will pick up on differences in values (for example, Sam versus Samuel), but it will also pick up on differences in formatting and capitalization, as illustrated in the preceding figure. Make sure that you understand and account for this when using the function.

FIND function

The **FIND** function enables the Excel user to find the position of one text string (like a word or series of letters) inside another text string.

Uses and applications

When dealing with addresses or other larger text strings, we often want to pull out a set piece of information, such as the city represented in an address. Doing this can be difficult because the starting position for that city may vary depending on the string itself. "FIND" lets us start to solve problems like this by determining the starting point for a given text string, as *figure 7.6* illustrates.

The formula

=FIND(text to be found, text to search, optional starting number for search)

If the starting number value is omitted, Excel will start the search at 1 or the first character in the cell.

Figure 7.6 illustrates the use of the **FIND** function.

	A	B	C
25			
26			
27	16	Hello	This cell says Hello
28	#VALUE!	Hello	Goodbye
29	=FIND(B28,C28)		

Figure 7.6: *The FIND function*

Errors to avoid

Notice that if the text to be found is not in the text string, the value returned will be #VALUE!, as shown in the example searching for the string "Hello" inside of "Goodbye."

FIXED function

The **FIXED** function creates a number with a fixed number of decimal places from another value.

Uses and applications

The **FIXED** function is essentially an alternative to using a standard formatting approach for a number. This is mainly useful when pulling in large amounts of data and then looking to display a snapshot of it in another part of Excel. **FIXED** takes a given number, rounds it to a given set of decimal places, and then either includes or excludes commas before returning the result as text.

The formula

=FIXED(number to be formatted, number of decimals to be shown, optional commas/no commas indicator)

Figure 7.7 illustrates the use of the **FIXED** function:

	A	B
31		
32	3.142	3.141593
33	=FIXED(B32,3,0)	

Figure 7.7: *The FIXED function*

Errors to avoid

The **FIXED** function is easy to use—just make sure you are referring to the number that you want to be formatted, and you should have no trouble.

LEFT and RIGHT functions

The **LEFT** and **RIGHT** functions allow Excel users to pull portions of text from another cell, starting from either the left or right side of the referenced value.

Uses and applications

In many datasets, we may need to pull a substring out of a larger dataset—for instance, pulling the postal code at the end of an address or pulling an individual's first name from a cell containing their whole name. In these circumstances, the **LEFT** and **RIGHT** functions can be especially useful. These functions can also be combined with the **FIND** function (using a space " " value to be found) to find the first or last word in a string.

The formula

=LEFT(text value to be drawn from, number of characters to pull starting on the left side)

=RIGHT(text value to be drawn from, number of characters to pull starting on the left side)

Figure 7.8 illustrates the use of the **LEFT** and **RIGHT** functions:

	A	B
35		
36	3.1	3.141593
37	=LEFT(B36,3)	
38	ello	Hello
39	=RIGHT(B38,4)	
40	15	3.141593
41	=LEFT(RIGHT(B40,4),2)	

Figure 7.8: *The LEFT and RIGHT functions*

Errors to avoid

LEFT and **RIGHT** are extremely useful functions, and they can be combined with other functions such as **FIND** or with one another (for example, pull the left three characters from a subset of the right seven characters in a cell). However, these kinds of nested functions can quickly get complicated, so be sure to check that the results you are getting back match what you expected in case of any errors in writing the formula.

LEN function

The **LEN** function is a simple function to return the length of a text string in another cell.

Uses and applications

LEN is useful mainly when combined with other functions such as **LEFT** and **RIGHT**. For instance, the function can be used to determine the length of a word or string in another cell, which can then be used to pull a fixed number of characters from that cell.

The formula

=LEN(cell where the number of characters should be counted)

Figure 7.9 illustrates the use of the **LEN** function:

44	5	Hello
45	=LEN(B44)	

Figure 7.9: *The LEN function*

Errors to avoid

Like several other text functions, the **LEN** function is easy to use—just make sure you are referring to the text that you want, and you should have no trouble.

LOWER, UPPER functions

The **UPPER** and **LOWER** functions convert all letters in another cell into either lower case or upper-case values.

Uses and applications

It is quite common for a dataset to have capitalization differences in different values in the dataset. For instance, with addresses, countries may have every letter capitalized, none of the letters capitalized, or all of the letters capitalized. Last names often have differences in capitalization between individuals. And, of course, any data pulled from information typed by users (such as filling in e-mail addresses) can have a wide variety of differences in capitalization. For this reason, it is often easiest to convert all values of a given variable in a dataset to either be upper or lower case—and this is most effectively done with the **UPPER** and **LOWER** functions. Note that this also helps avoid issues with the **EXACT** function and capitalization differences between cells.

The formula

=UPPER(Cell reference to be converted to upper case letters)

=LOWER(Cell reference to be converted to lower case letters)

Figure 7.10 illustrates the use of the **UPPER** and **LOWER** functions:

	A	B
47		
48	HELLO	hElLo
49	=UPPER(B48)	
50	hello	hElLo
51	=LOWER(B50)	

Figure 7.10: *The UPPER and LOWER functions*

Errors to avoid

The **UPPER** and **LOWER** functions are easy to use—just make sure you are referring to the correct cell, and you are unlikely to have any problems.

MID function

The **MID** function is essentially a more powerful version of **LEFT** or **RIGHT**, enabling you to pull a set of characters from a string at any starting point in that string.

Uses and applications

Although nested **LEFT** and **RIGHT** functions can be used to pull specific sets of characters from the middle of words or strings in a cell, **MID** is an easier way to do this. For instance, if one wants to try to pull a street name in an address, the **MID** function is an effective way to do that.

The formula

=MID(cell reference for the text to be pulled, starting point in the reference string, number of characters to pull from the starting point in reference string)

Figure 7.11 illustrates the use of the **MID** function:

54	ee	Greetings
55	=MID(B54,3,2)	
56	reet	Greetings
57	=MID(B56,2,4)	

Figure 7.11: *The MID function*

Errors to avoid

The **MID** function is very powerful, but it is also easy to get the function arguments mixed up—for instance, putting the number of characters to be pulled in place of the starting point argument—be sure to check the output to make sure it is returning the desired result.

NUMBERVALUE function

NUMBERVALUE is a nifty function that reads the value from another cell and converts that value to a number, ignoring any spaces in the digits. Of course, this is only possible if the text is a number, so non-numeric values will return an invalid value.

Uses and applications

Data pasted into Excel is often read as text values, even when it is numbered, and will frequently contain errant spaces and other formatting issues. The **NUMBERVALUE** function solves many of these issues—it ignores any spaces in the number and converts numeric digits to number values rather than text values.

The formula

=NUMBERVALUE(numeric text to be converted to a number).

Figure 7.12 illustrates the use of the **NUMBERVALUE** function:

	A	B
58		
59	1305.6789	13 05.6789
60	=NUMBERVALUE(B59)	

Figure 7.12: *The NUMBERVALUE function*

Errors to avoid

The **NUMBERVALUE** function will only work with numeric digits—any standard words or other non-numeric characters will just return **#VALUE!**

PROPER function

Dealing with proper nouns—people's names, cities, countries, and so on—can be a hassle, but the **PROPER** function in Excel makes it much easier.

Uses and applications

Data values for proper nouns are often in all lower case, or all upper case letters, but standard written form dictates that these words should have a capitalized first letter, and then all subsequent letters should be lower-case. This formatting can be easily applied to any cell with the **PROPER** function, as *figure 7.12* shows.

The formula

=PROPER (Text or cell reference to be formatted as a proper noun)

Figure 7.13 illustrates the use of the **PROPER** function:

63	London	lonDon
64	=PROPER(B63)	

Figure 7.13: *The PROPER function*

Errors to avoid

The **PROPER** function is quite simple, but make sure that you are careful if using it in a nested function, in combination with a function such as **LEFT** or **MID**. In these circumstances, getting the formula right can be tricky.

REPLACE function

Microsoft's *Ctrl + H* option is a popular choice for replacing values in large quantities in a dataset, but the **REPLACE** function in Excel can serve a similar purpose as well and is more dynamic than a simple *Ctrl + H*.

Uses and applications

Sometimes in a dataset, we may need to replace a series of old values with new ones, such as updating a list of addresses or locations, changing employee information, or replacing outdated values with new ones. While we could use Control + H to replace the same value everywhere with a new one, in database work, it is common to need to replace certain old values with a new one that differs by row (for example, updating office numbers for each employee in a list). **REPLACE** can do this effectively.

The formula

=REPLACE(cells with old text to be replaced, starting position for the text to be replaced, number of characters to be replaced, the new text to be used as a replacement)

Figure 7.14 illustrates the use of the **REPLACE** function. The starting position for the text to be replaced is the number of characters from the first character in the cell to begin replacing text. In the following example, we start at character 14, which is where the "O" in Old is. Then the number of characters to be replaced is the number of characters we want to overwrite—in this case, it is the three characters of Old being overwritten with New.

68	This Text is New	This Text is Old
69	=REPLACE(B68,14,3,B69)	New

Figure 7.14: *The REPLACE function*

Errors to avoid

The hardest aspect of using the **REPLACE** function is identifying the starting position for the text to be replaced (**SEARCH** or **FIND** can help with this), so be sure to check that the replacement is starting in the proper position.

SEARCH function

SEARCH returns the location of a substring or value from another text string in a cell.

Uses and applications

SEARCH is best combined with other functions, such as **REPLACE** or **RIGHT**, for instance, to create effective nested functions. It enables the user to read a complex string and identify the location of a particular value. For instance, the example in *figure 7.14* searches a short sentence for the position of the first space in the sentence.

The formula

=SEARCH(value to be found, value to be searched)

Figure 7.15 illustrates the use of the **SEARCH** function:

73	6	Where is the first space?
74	=SEARCH(" ",B73)	
75	1	Where is the w?
76	=SEARCH("w",B75)	
77	14	Where is the w?
78	=FIND("w",B77)	

Figure 7.15: *The SEARCH function*

Errors to avoid

The **SEARCH** function is functionally similar to the **FIND** function—the major difference is that **FIND** is case-sensitive, whereas **SEARCH** is not. As a result, the two functions can produce different values, as the preceding figure shows.

SUBSTITUTE function

The **SUBSTITUTE** function is primarily useful when you have pulled in large amounts of text data and want to remove particular values, similar to the **REPLACE** function, but only in given locations in the data.

Uses and applications

SUBSTITUTE is similar to **REPLACE**, but with an important difference; **REPLACE** will replace any text at a specific point in a text string. In contrast, **SUBSTITUTE** replaces a specific set of text in a string, regardless of where it occurs. *Figure 7.15* illustrates this difference. Basically, **SUBSTITUTE** is useful for replacing a specific value in multiple instances where it occurs, whereas **REPLACE** is better for dynamically changing specific values that might differ for different rows in a dataset.

The Formula

=SUBSTITUTE(Original text, value to be substituted, new text to be inserted as substitute for old)

Figure 7.16 illustrates the use of the SUBSTITUTE function:

81	This Text is New	This Text is Old
82	=REPLACE(B81,14,3,B82)	New
83	This New Text is New	This Old Text is Old
84	=SUBSTITUTE(B83,"Old",B84)	New

Figure 7.16: *The SUBSTITUTE function*

Errors to avoid

The **SUBSTITUTE** function is easy to mix up with **REPLACE**—but **SUBSTITUE** will change **ALL** instances of the text to be changed. This makes **SUBSTITUTE** less precise than **REPLACE**—be sure you know which one you want to use for the specific use case.

T function

The T function reads a value in another cell and either return that value if it is text or returns a blank cell if the value is not text.

Uses and applications

The T function is a simple Excel function that can be used to screen out non-text values from a dataset. It is an easy and quick way to clean a mixed dataset where numbers and text values are mixed together.

The formula

=T(Text to be cleaned, or reference cell containing the text)

Figure 7.17 illustrates the use of the T function:

87	A B C D E F G	A B C D E F G
88		123456
89	=T(B88)	

Figure 7.17: *The CLEAN function*

Errors to avoid

The T function is easy to use—just make sure you understand what it does, and you are unlikely to make an error.

TEXT function

The **TEXT** function allows an Excel user to read a value from another cell and convert it to a particular format.

Uses and applications

TEXT is primarily useful for cleaning up large amounts of raw data. Often, data imported into Excel is not in the format the user wants, and Excel allows the user to quickly fix that issue with the **TEXT** function. The key caveat is that Excel requires the format to be specified in the function's formula, and those formats are not intuitive. *Figure 7.17* illustrates four common formats for currency values, dates and days of the week, and percentages.

The formula

=CLEAN(Text to be cleaned, format for the text)

Figure 7.18 illustrates the use of the **TEXT** function:

	A	B
90		
91	$9,876.54	=TEXT(9876.543,"$#,##0.00")
92	11/17/22	=TEXT(TODAY(),"MM/DD/YY")
93	Thursday	=TEXT(TODAY(),"DDDD")
94	12.5%	=TEXT(0.125,"0.0%")

Figure 7.18: *The TEXT function*

Errors to avoid

The **TEXT** function is easy to use—once you know the format, you want for the values to be cleaned. So be sure to check that your format works and produces the desired output before you apply it to a list of cells.

TRIM function

The **TRIM** function is a simple function to remove excess spaces between words in Excel.

Uses and applications

As with many other text functions, the primary purpose of the **TRIM** function is to clean up a large dataset that has been imported to Excel. In such cases, there are often many unneeded spaces or tabs between values or words. **TRIM** removes these and leaves only a single space between words.

The formula

=TRIM(text to be cleaned)

Figure 7.19 illustrates the use of the **TRIM** function:

97	This sentence is messy.	This	sentence is	messy.
98	This sente nce is m ess ier.	This	sente nce is	m ess ier.
99	=TRIM(B98)			

Figure 7.19: *The TRIM function*

Errors to avoid

Note that Excel does not actually know what is or is not a word—so for spaces that occur within a single work, **TRIM** will not be effective in removing the extra spaces. This is illustrated in the second row in the preceding figure, where the words "sentence" and "messier" have extraneous spaces in them—Excel can remove the double spaces between words but not the unneeded spaces in words themselves.

UNICHAR function

The **UNICHAR** function converts numbers into the associated Unicode character represented by that number.

Uses and applications

In some cases, an Excel user may need to convert from a series of numbers into the associated Unicode characters—in this case, **UNICHAR** is an effective way to handle the problem.

The formula

=UNICHAR(number to be converted to a Unicode character)

Figure 7.20 illustrates the use of the **UNICHAR** function:

105	B	66
106	=UNICHAR(B105)	
107	J	74
108	=UNICHAR(B107)	
109	Q	81
110	=UNICHAR(B109)	

Figure 7.20: *The UNICHAR function*

Errors to avoid

The **UNICHAR** function is easy to use—as long as you have the correct numbers input in the sheet. Check your output to make sure you do not have any errors in the numeric values.

UNICODE function

UNICODE returns the numeric value associated with a specific Unicode character.

Uses and applications

UNICODE is essentially the opposite of **UNICHAR**. The **UNICODE** function takes a letter (or the first letter in a word) and returns the number associated with that letter, as *figure 7.20* illustrates.

The formula

=UNICODE(text to be converted to numeric value)

Figure 7.21 demonstrates the **UNICODE** function.

	A	B	C
112			
113	65	A	=UNICODE(B113)
114	66	Boy	=UNICODE(B114)
115	67	Cat	=UNICODE(B115)
116	68	Dog	=UNICODE(B116)
117	69	Even So	=UNICODE(B117)
118	70	F	=UNICODE(B118)
119	71	G	=UNICODE(B119)

Figure 7.21: *The UNICODE function*

Errors to avoid

The **UNICODE** function is very easy to use. Again, just remember that when referencing multiple letters or words in a cell, **UNICODE** will only return the value for the first letter in that cell.

VALUE function

The **VALUE** function converts a text string of a number into a numeric value in Excel.

Uses and applications

Text strings in Excel are treated like text, even if they are numbers. Yet, often for database work, these values need to be in a numeric format. To help deal with that issue, the **VALUE** function can be used to turn string values into numeric values.

The formula

=VALUE(text containing string value to be converted to number format)

Figure 7.22 demonstrates how **VALUE** is used.

123		2	2
124	=VALUE(B123)		
125		3	3
126	=VALUE(B125)		

Figure 7.22: *The VALUE function*

Errors to avoid

The **VALUE** function is quite simple to use—it is unlikely you will run into any problems with it.

Conclusion

The functions shown in this chapter are text functions which are mainly helpful in cleaning up data that is imported into Excel or in maintaining and updating a database. They are generally easy to use but are also major time savers if you learn them.

Points to remember

There are a variety of text functions that help us clean and organize datasets. The most important ones to remember are as follows:

- **CONCATENATE function:** When you want to combine two text strings together into a single value, for instance, a first and last name in separate cells, Excel's **`CONCATENATE`** function is your best option.
- **FIND function**: The **`FIND`** function enables the Excel user to find the position of one text string (like a word or series of letters) inside another text string.
- **LEFT** and **RIGHT functions:** The **`LEFT`** and **`RIGHT`** functions allow Excel users to pull portions of text from another cell, starting from either the left or right side of the referenced value.
- **UPPER** and **LOWER functions:** The **`UPPER`** and **`LOWER`** functions convert all letters in another cell into either lower-case or upper-case values.
- **REPLACE functions:** Microsoft's *Ctrl+H* option is a popular choice for replacing values in large quantities in a dataset. However, the **`REPLACE`** function in Excel can serve a similar purpose as well and is more dynamic than a simple *Ctrl+H*.
- **SEARCH functions: `SEARCH`** returns the location of a substring or value from another text string in a cell.
- **SUBSTITUTE function:** The **`SUBSTITUTE`** function is primarily useful when you have pulled in large amounts of text data and want to remove particular values, similar to the **`REPLACE`** function, but only in given locations in a string.

Multiple choice questions

1. If you wanted to replace the word "Hello" with the word "Greetings" in a series of text strings, but only at a specific point in each string, which function would be most effective for the task?
    a. REPLACE
    b. SUBSTITUTE

   c. TEXT
   d. CLEAN

2. You have a series of names in a column in Excel, with each cell formatted as "Last Name, First Name." If you want to create a separate column that contains only the first three letters of each person's last name, which formula would you use?
   a. RIGHT (Reference Cell, 3)
   b. LEFT (Reference Cell, 3)
   c. CLEAN (LEFT [3, Reference Cell])
   d. CLEAN (RIGHT [3, Reference Cell])

3. A human resources department is trying to update its database in an Excel sheet. Column A contains a series of names for individuals, and Column B contains work departments for those individuals. If the company wants to combine both columns into a single column in the format "Name, Work Department," what formula could they use?
   a. UPPLER
   b. PROPER
   c. TEXT
   d. CONCATENATE

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 8

Information Functions in Excel

Introduction

Like text functions, information functions can also help us to go through a large dataset and quickly clean it up to make it usable for our work. In this chapter, we will cover the most popular and useful formulas you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- CELL function
- ERROR.TYPE function
- INFO function
- ISBLANK function
- ISERROR and ISERR functions
- ISEVEN and ISODD functions
- ISFORMULA function

- ISLOGICAL function
- ISNA function
- ISNUMBER function
- ISREF function
- ISTEXT function
- N function
- SHEET function
- TYPE function

Objectives

After studying this unit, the reader will be able to apply Information functions to Excel workbooks.

Cell function

The **CELL** function returns information about another cell in the Excel workbook, such as the format of that cell or its row number.

Uses and applications

The **CELL** function is primarily used to provide information about another cell in the workbook, usually as part of a broader formula to accomplish a task in Excel, such as displaying a value if a given cell is formatted in a particular way. Although **CELL** itself is just one function, there are about a dozen different characteristics Excel can check for in the reference cell.

The formula

=CELL(info type, reference cell or cells)

Here, info type refers to one of the characteristics Excel can check for in a reference cell, including Address (cell address), Col (column), color (an indicator variable if a cell has a red value when a negative number is an input), contents (value of the upper left cell in the reference), filename (file path of the cell as text), format (text value denoting the format of the cell), parentheses (indicator variable for whether negative numbers are formatted to have parentheses), prefix (indicates how text is aligned in the cell—left, right, or center), protect (indicator variable if the cell is locked or protected from editing), row (the row of the cell), type (indicates what

content is in the cell—blank, text, or something else), and width (column width of the cell and whether it is default width or adjusted by the user).

Figure 8.1 illustrates the use of the **CELL** function with a few of the possible characteristics shown:

	A	B	C
1	$B$1	ABC	=CELL("address",B1)
2	1	($12)	=CELL("color",B2)
3	F3	3.142	=CELL("format",B3)

Figure 8.1: *The CELL function*

Errors to avoid

The **CELL** formula is fairly complex. There are two likely areas for a mistake. First, note that the characteristic to be returned has to be in quotation marks—"XYZ"—and second, there are a lot of characteristics that can be used, so make sure you know which one you want and what the output returned will be from it.

ERROR.TYPE function

The **ERROR.TYPE** function is a handy tool to enable the user to identify and code any errors that occur in Excel

Uses and applications

It is very common for Excel to have errors coming from a function. The most common is probably #N/A!, but there are actually eight possible error types that can occur, and with a large mass of formulas in a dataset, you may want to be able to determine which errors are coming up in order to deal with them. That is exactly what the **ERROR.TYPE** function does.

The formula

=ERROR.TYPE(Cell reference)

Figure 8.2 illustrates the use of the **ERROR.TYPE** function and the eight different types of errors that can arise. The left column contains the formulas and reads the errors

that have occurred in column B. For each type of error, a different number value is returned.

6	1	#NULL!	=ERROR.TYPE(B6)
7	2	#DIV/0!	=ERROR.TYPE(B7)
8	3	#VALUE!	=ERROR.TYPE(B8)
9	4	#REF!	=ERROR.TYPE(B9)
10	5	#NAME?	=ERROR.TYPE(B10)
11	6	#NUM!	=ERROR.TYPE(B11)
12	7	#N/A	=ERROR.TYPE(B12)
13	8	#GETTING_DATA	=ERROR.TYPE(B13)

Figure 8.2: *The ERROR.TYPE function*

Errors to avoid

The **`ERROR.TYPE`** function is easy to use—but beware, if the reference cell does not contain an error, then **`ERROR.TYPE`** itself will return the #N/A! error; perhaps Excel's attempt at ironic humor?

INFO function

The **`INFO`** function is a nifty Excel function that returns information about the computer that is operating Excel.

Uses and applications

INFO is a surprising function for many people—it essentially pulls information about the computer that is operating Excel and then displays that information in Excel. The surprising thing about it is that many people do not realize Excel has the capability to do things like determining the operating system in use on the computer or the system parameters of the software.

The formula

=INFO(type of information to be returned)

Figure 8.3 illustrates the use of the **`INFO`** function with a few of the possible characteristics of the computer system that can be returned.

	A	B
15		
16	pcdos	=INFO("system")
17	$A:$A$15	=INFO("origin")
18	Windows (32-bit) NT :.00	=INFO("Osversion")
19	Automatic	=INFO("recalc")

Figure 8.3: *The INFO function*

Errors to avoid

`INFO` is an easy function to use—just be sure that you specify a characteristic, Excel will return, and that you do so in quotes such as "system."

ISBLANK function

The **`ISBLANK`** function is a simple function that returns a TRUE/FALSE value if a cell is blank or not.

Uses and applications

For simple Excel applications, it is easy to determine visually if a cell is blank or not and take the appropriate action. For large datasets or complex Excel models, though an automated method of determining if a cell is blank or not, can be handy. Enter the **`ISBLANK`** function.

The formula

=ISBLANK(reference cell)

Figure 8.4 illustrates the use of the **`ISBLANK`** function:

21	TRUE	
22	FALSE	=ISBLANK(B22)

Figure 8.4: *The ISBLANK function*

Errors to avoid

`ISBLANK` is an easy function to use—as long as you get the correct reference cell or use it properly in a larger formula, and you should not have any issues.

ISERROR and ISERR functions

ISERROR and **ISERR** are two Excel functions that both fulfill the same basic purpose—they return a value of true or false to determine if a particular cell contains an error.

Uses and applications

For complex Excel models, rather than going and manually looking to see if a particular cell contains an error value, the **ISERROR** and **ISERR** functions can be more efficient. The functions are essentially identical, except that **ISERR** will not pick up the very common #N/A (not available) values as errors, whereas **ISERROR** will. In some cases, we may wish to treat #N/A as an error, and in other cases, we may not—hence, the two formulas.

The formula

=ISERROR(reference cells)

= ISERR(reference cells)

Figure 8.5 illustrates the use of the **ISERR** and **ISERROR** functions—notice that neither function returns a **TRUE** value for the error if the cell is blank, but both will pick up #NUM! as an error. Only **ISERROR** will treat #N/A as an error.

25	FALSE		=ISERROR(B25)
26	FALSE		=ISERR(B26)
27	TRUE	#N/A	=ISERROR(B27)
28	FALSE	#N/A	=ISERR(B28)
29	TRUE	#NUM!	=ISERROR(B29)
30	TRUE	#NUM!	=ISERR(B30)

Figure 8.5: *The ISERR and ISERROR functions*

Errors to avoid

ISERR and **ISERROR** are easy to use—but do not forget that **ISERR** will not treat #N/A values as errors.

ISEVEN and ISODD functions

ISEVEN and **ISODD** are simple functions that return a true or false value when a number is even or odd.

Uses and applications

In complex formulas, it can be helpful to know if a value in another cell is an even number or an odd one. This could come up for mathematical operations or when numbers are being used to classify data into particular numbered categories.

The formula

=ISEVEN(reference cells)

=ISODD(reference cells)

Figure 8.6 illustrates the use of the **ISEVEN** and **ISODD** functions:

34	FALSE	11	=ISEVEN(B34)
35	TRUE	11	=ISODD(B35)
36	TRUE	22	=ISEVEN(B36)
37	FALSE	22	=ISODD(B37)

Figure 8.6: *The ISEVEN and ISODD functions*

Errors to avoid

ISEVEN and **ISODD** are simple functions to use—always check that the reference cell is correct, but otherwise, you are unlikely to have any issues.

ISFORMULA function

Excel has a series of "IS" type functions that return a TRUE/FALSE value based on the contents of another cell. **ISFORMULA** is an example of that type of function

Uses and applications

ISFORMULA is mainly useful in a larger formula or an Excel model. It can be used to evaluate whether a cell is a formula or not, and then based upon that, take some action.

The formula

=ISFORMULA(Reference cell)

Figure 8.7 illustrates the use of the **ISFORMULA** function. The function returns FALSE when referring to the blank cell in row 41 and **TRUE** when referring to the formula in row 40.

40	TRUE	5	=ISFORMULA(B40)
41	FALSE		=ISFORMULA(B41)

Figure 8.7: *The ISFORMULA function*

Errors to avoid

Many of the information functions are simple to apply, and **ISFORMULA** is no exception—be careful to select the correct reference cell, but otherwise, you should have no issues.

ISLOGICAL function

Similar to the **ISFORMULA** function, **ISLOGICAL** enables the user to check if a cell is a "logical value"—in other words, true or false.

Uses and applications

ISLOGICAL is primarily useful for testing large amounts of cells and removing those where the output does not fit certain criteria—for instance, a formula might check if an HR applicant is a college graduate or not and return a TRUE/FALSE value. The **ISLOGICAL** function can then be used to remove (or at least flag) all values where there was no indication of education for the applicant, and hence, the TRUE/FALSE question cannot be answered, or the cell contains an error.

The formula

=ISLOGICAL(reference cell)

Figure 8.8 illustrates the use of the **ISLOGICAL** function.

44	TRUE	TRUE	=ISLOGICAL(B44)
45	FALSE	XYZ	=ISLOGICAL(B45)

Figure 8.8: *The ISLOGICAL function*

Errors to avoid

Again, **ISLOGICAL** is simple to apply, so other than a mistake in cell references, you are unlikely to make any errors when using it directly.

ISNA function

The **ISNA** function, like other **IS** functions, tests a cell for a specific condition—in this case, whether the cell contains one of the most common types of errors—the value #N/A meaning "Not Applicable."

Uses and applications

#N/A is one of the most common Excel errors and comes up often when doing calculations across large numbers of cells in Excel. The **ISNA** function can effectively flag those cells with an error, enabling the user to address the error as needed.

The formula

=ISNA(reference cells)

Figure 8.9 illustrates the use of the **ISNA** function:

47	FALSE	XYZ	=ISNA(B47)
48	FALSE	NA	=ISNA(B48)
49	TRUE	#N/A	=ISNA(B49)

Figure 8.9: *The ISNA function*

Errors to avoid

The **ISNA** function is easy to use, but notice that it only picks up the specific syntax associated with the Not Applicable error—#N/A. As the preceding figure shows, it will not detect the text letters NA.

ISNUMBER function

ISNUMBER is a simple handy function that detects whether an Excel function contains a number.

Uses and applications

The **ISNUMBER** function is primarily useful for detecting data entry errors—spots in an Excel sheet where a numeric value should have been used, but some other type of value ended up there instead. For instance, a column might contain student's ages, but in one case, perhaps that value is missing, or a student's name was accidentally entered—ISNUMBER can detect these types of issues.

The formula

=ISNUMBER(reference cell)

Figure 8.10 shows how to use the **ISNUMBER** function

51	FALSE	Hello	=ISNUMBER(B51)
52	TRUE	1	=ISNUMBER(B52)
53	TRUE	3.1415	=ISNUMBER(B53)
54	FALSE	08:00AM	=ISNUMBER(B54)

Figure 8.10: *The ISNUMBER function*

Errors to avoid

ISNUMBER is one of the easiest functions in Excel to use—check your references, and you should not have any issues with it.

ISREF function

ISREF is a bit different than the other IS functions—while most IS functions evaluate whether a reference cell meets a specific condition, **ISREF** instead evaluates whether the text contained in the function leads to a valid cell.

Uses and applications

ISREF is only likely to be used in a more complex function or formula—for instance, another function might generate a cell location or address, and then ISREF can be used to test whether that cell address is valid.

The formula

=ISREF(reference address to be checked)

Figure 8.11 illustrates **ISREF** in use.

56	TRUE	=ISREF(A1)
57	TRUE	=ISREF(AB57)
58	FALSE	=ISREF(ABC58)
59	FALSE	=ISREF(HELLO)

Figure 8.11: *The ISREF function*

Errors to avoid

Remember, **ISREF** does NOT check whether a reference exists at another cell—it checks whether the text in question is a valid cell reference or address. In the preceding figure, cell AB57 is blank, but the **ISREF** function still returns TRUE because cell AB57 does exist in the sheet. In contrast, cell ABC58 does not exist, so **ISREF** returns FALSE.

ISTEXT function

ISTEXT is a bit like **ISNUMBER**, but the opposite—**ISNUMBER** detects numbers in a reference cell, and **ISTEXT** detects **TEXT**.

Uses and applications

ISTEXT is less useful than **ISNUMBER**, mainly because Excel is used for a lot more computations than text database work. However, **ISTEXT** is still handy to have whenever you have a large dataset and want to make sure that certain variables are all of a particular type—in this case, text values, as *figure 8.12* shows.

The formula

=ISTEXT(reference cell)

Figure 8.12 shows how to use the **ISTEXT** function

61	TRUE	Hello	=ISTEXT(B61)
62	FALSE	1	=ISTEXT(B62)
63	FALSE	3.1415	=ISTEXT(B63)
64	TRUE	08:00AM	=ISTEXT(B64)

Figure 8.12: *The ISTEXT function*

Errors to avoid

Notice that time values are treated as **TEXT** and not numbers in Excel. Be careful to be aware of this if working with times in a dataset.

N function

The **N** function converts non-numeric values such as dates and true or false values to numbers.

Uses and applications

N is a surprisingly useful function for two separate types of applications. First, when dealing with dates, it is often messy to calculate something like the number of days between two dates. The **N** function makes that easy to do by converting date values to serial numbers, which can then be added or subtracted as needed.

Second, many Excel functions, such as the **IS** functions, produce true/false outputs, but the user may want to know how common those true or false values are. For instance, **ISNA** tells us whether a cell contains an #N/A error—**N** can then be used to turn **TRUE** output from that function into values of 1, which can then be summed up to determine the number of N/A issues in the dataset.

The formula

=N (reference cell)

Figure 8.13 demonstrates the use of **N** in a number of situations. Note that **N** will produce a serial number for dates, a 1 for a value of **TRUE**, and a 0 for any other non-numeric value, including a blank cell. Numeric values are returned as themselves again.

67	1	TRUE	=N(B67)
68	0	FALSE	=N(B68)
69	43831	1/1/2020	=N(B69)
70	0	HELLO	=N(B70)
71	3.1415	3.1415	=N(B71)
72	0		=N(B72)

Figure 8.13: *The N function*

Errors to avoid

The N function is easy to use, but be sure you realize that **N** will return the value 0 even for blank cells.

SHEET function

The **SHEET** function is a quick function to return the number of a cell in a particular worksheet within Excel.

Uses and applications

In more complex Excel models, you may need to use multiple worksheets within the Excel workbook, and you may be using formulas that need to reference formulas

across these different worksheets. In this case, the **SHEET** function returns a numeric value that corresponds to the worksheet in question.

The formula

=SHEET(reference cell)

Figure 8.14 demonstrates how the sheet function works. The first row refers to a cell in the first worksheet, and so the number 1 is returned. The next row refers to a cell in sheet 2, so the number 2 is returned. Please refer to the following figure:

	A	B	C
68			
69	1	Hello	=SHEET(B69)
70	2		=SHEET(Sheet2!A4)

Figure 8.14: *The SHEET function*

Errors to avoid

The **SHEET** function is easy to use—but remember that Excel simply pulls these values based on the order of the sheets in the workbook. So if you change the position of the sheets—for instance, moving one forward in front of others, the **SHEET** function value will also change. **SHEET** does not number worksheets based on when they were created but instead based on the order they appear in.

Type function

The **TYPE** function is a simple function that classifies values in Excel cells and returns a number depending on what type the value is.

Uses and applications

You can use **TYPE** when dealing with large amounts of data to classify and then sort values for a variety of purposes. For instance, if we want to identify all of the different types of errors in a dataset, we could use **TYPE** to classify all of the cells and then sort the resulting rows by the **TYPE** value, and all errors would appear under the value 16 as shown in *figure 8.15*. Numbers have the **TYPE** of 1, whereas logical values are **TYPE 4**, and text values are **TYPE 2**.

The formula

=TYPE (reference cells)

Figure 8.15 illustrates the use of the **TYPE** function:

74	1	3.1415
75	2	Hello
76	4	TRUE
77	16	#N/A

Figure 8.15: *The TYPE function*

Errors to avoid

The **TYPE** function is easy to use—just check your reference cells for accuracy, and you should not run into any issues.

Conclusion

Information functions can help you build more effective Excel models, particularly if you have a lot of messy data that you need to clean up and make useable. The functions shown in this chapter are a great starting point in building effective databases. They are generally easy to use, but are also major time savers if you learn them. Excel is not the software that most people prefer for intensive database development – and that is fair, it lacks many core features for big database work – but, Excel can still be a useful and practical first step in building and visualizing dataset before transitioning the data over to a more specialized software program. This is how I, and many other data scientists, use Excel in regular practice for building data sets.

Points to Remember

While there are a number of information functions, the most important ones to remember are as follows:

- **CELL function:** The **CELL** function returns information about another cell in the Excel workbook, such as the format of that cell or its row number.
- **INFO function:** The **INFO** function is a nifty Excel function that returns information about the computer that is operating Excel.
- **ISBLANK function:** The **ISBLANK** function is a simple function that returns a TRUE/FALSE value if a cell is blank or not.
- **ISERROR and ISERR function:** **ISERROR** and **ISERR** are two Excel functions that both fulfil the same basic purpose—they return a value of true or false to determine if a particular cell contains an error.

- **N function:** The `N` function converts non-numeric values such as dates and true or false values to numbers.
- **TYPE function:** The `TYPE` function is a simple function that classifies values in Excel cells and returns a number depending on what type the value is.

Multiple choice questions

1. **An Excel user wants to build a model that will display information differently to users of Linux and Windows operating systems. What function could be used to determine which system is being used to run Excel?**
   a. SYSTEM
   b. INFO
   c. TEXT
   d. TYPE

2. **You have a large dataset and want to start to clean it by removing all blank cells and values. Which function could be used to determine if a cell is blank?**
   a. ISNA
   b. ISBLANK
   c. CLEAN
   d. SPACE

3. **You have a large set of data with mixed types of values—some numbers, some text values, and some formulas. Which function would let you go through and categorize these values?**
   a. ISNUMBER
   b. PROPER
   c. INFO
   d. TYPE

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

CHAPTER 9
Financial Functions in Excel

Introduction

One of the most common uses of Excel is financial modeling. Excel is an excellent tool for financial forecasting, keeping track of expenses and income, doing valuation calculations, and much more. To help with financial work, Excel comes with a large set of finance functions. In this chapter, we will cover the most popular and useful financial formulas that you need to know to be able to use Excel effectively.

Structure

In this chapter, we will cover the following topics:

- COUPDAYS, COUPNUM, and COUPPCD functions
- CUMIPMT and CUMPRINC functions
- DB and DDB functions
- DISC function
- DURATION and MDURATION functions
- EFFECT function

- Time value of money function
    - PV function
    - FV function
    - NPER function
    - RATE function
    - PMT function
- INTRATE function
- IPMT function
- IRR, MIRR, and NPV functions
- NOMINAL function
- PRICE, YIELD, YIELDDISC, and YIELDMAT functions
- XIRR and XNPV functions
- VDB function

Objectives

After studying this unit, you will be able to apply financial functions to Excel workbooks.

COUPDAYS, COUPNUM, and COUPPCD functions

Excel's COUP functions, including **COUPDAYS**, **COUPNUM**, and **COUPPCD**, are designed to make bond calculations around coupon payments easier.

Uses and applications

The **COUP** functions give Excel users tools for bond calculations. All three functions enable calculations related to the payments that are due to bondholders. In particular, **COUPDAYS** calculates the number of days in a coupon period, **COUPNUM** calculates the number of coupon payments that should be made between the time when a bond is bought and the maturity date, and finally, **COUPPCD** calculates the prior coupon date from when the bond was most recently purchased (the settlement date).

The formula

=COUPDAYS(Settlement date, Maturity date, Frequency, Basis)

=COUPNUM(Settlement date, Maturity date, Frequency, Basis)

=COUPPCD(Settlement date, Maturity date, Frequency, Basis)

All three coupon formulas use the same basic set of inputs. The settlement date is the date that the bond purchase is finalized (for example, settled). The maturity date is the date that the bond will mature (which must be after the settlement date). The frequency is the number of periods for the bond per year: 1 for bonds that pay an annual coupon, 2 for semi-annual coupon bonds, and 4 for bonds with a quarterly coupon. The Basis is the days' convention being used with the bond: actual number of days, 365 days per year, 360 days per year, and so on. In Excel, it is expressed as an indicator variable ranging from 0 to 4, but it is also optional in the formula with a default setting of 30/360 on a days' Basis based on **NASD** standards.

Figure 9.1 illustrates the three **COUP** functions:

	A	B
1	Settlement Date	10/15/2023
2	Maturity Date	1/25/2025
3	Periods Per Year	2
4	Basis	1
5		
6	184	=COUPDAYS(B1,B2,B3,B4)
7	3	=COUPNUM(B1,B2,B3,B4)
8	7/25/2023	=COUPPCD(B1,B2,B3,B4)

Figure 9.1: *The COUPDAYS, COUPNUM, and COUPPCD functions*

Errors to avoid

Two issues can come up when using the **COUP** functions: first, be careful to make sure that the settlement and maturity values go into Excel as date values, not text or numeric values (use **DATE()** if needed); second, the output from **COUPPCD** may be a date serial number (for example, 245678). If the latter occurs, change the format on the cell so that it is a date format, and that should produce a readable date as the output, as seen in the preceding *figure 9.1*.

CUMIPMT and CUMPRINC functions

Excel has two functions that are useful for calculating principal and interest payments on amortized loans: **CUMIPMT** and **CUMPRINC**.

Uses and applications

When dealing with amortized loans, calculating the payment amount is fairly straightforward—the PMT function is quick and easy to use. However, in some cases, we may want to calculate the cumulative amount of principal or interest that is paid over some period or portion of the loan. Doing this calculation generally requires building an amortization table and then summing up interest or principal payments for the period in question, or we can just use **CUMIPMT** (for interest payments) and **CUMPRINC** (for principal payments).

The formula

=CUMIPMT(rate, number of periods, PV, starting period, ending period, type)

=CUMPRINC(rate, number of periods, PV, starting period, ending period, type)

Both functions use the same basic inputs: the interest rate, the number of periods in the loan, the present value or initial amount of the loan, the starting period and ending periods to determine the amount of principal or interest paid, and then the "type" of the loan—that is whether payments are made at the beginning or end of each period (optional and set to end of period by default).

Figure 9.2 demonstrates how these functions are used:

10								
11	Start Period	1		Start Period	32			
12	End Period	1		End Period	36			
13	Rate	5.20%		Rate	5.20%			
14	Number of Periods	36		Number of Periods	36			
15	PV	1000		PV	1000			
16								
17	-52	=CUMIPMT(B13,B14,B15,B11,B12,0)		-43.04668148	=CUMIPMT(E13,E14,E15,E11,E12,0)			
18	-9.995198446	=CUMPRINC(B13,B14,B15,B11,B12,0)		-266.9293107	=CUMPRINC(E13,E14,E15,E11,E12,0)			

Figure 9.2: *The CUMIPMT and CUMPRINC functions*

Errors to avoid

It may be tempting to put in PV as a negative number, expecting to get a positive output for either formula. However, doing this will simply produce a #NUM error. PV must be negative for the function to work properly, and the output from both formulas will always be negative, consistent with a payment/negative cash flow for the borrower.

DB and DDB functions

Calculating depreciation on assets is something Excel makes easier with the DB and DDB functions. These functions give the accumulated depreciation under the declining balance and double declining balance methods.

Uses and applications

When dealing with large capital expenditures, businesses typically need to depreciate these assets over time to reflect their long-life nature rather than expensing the full cost in a single year. The **DB** and **DDB** functions offer an easy way to determine the depreciation that has occurred with an asset over a period of time.

The formula

=DB(Cost, Salvage, Life, Period, Month)

=DDB(Cost, Salvage, Life, Period, Month)

The depreciation functions use the same inputs but produce different outputs depending on whether the **declining balance** (**DB**) or **double declining balance** (**DDB**) method is used. The inputs are cost, salvage, life, period, and month. Cost is the cost of the capital equipment in question, salvage is the value at the end of its life, life is the expected life of the equipment (based on standard depreciation tables), Period is the period at which depreciation amount should be calculated, and month is an optional value for the month of the year when the calculation is being done.

Figure 9.3 illustrates the use of the two functions.

20	Cost	5000
21	Salvage	300
22	Life	5
23	Period	2
24	Month	3
25		
26	$1,918.88	=DB(B20,B21,B22,B23,B24)
27	$1,200.00	=DDB(B20,B21,B22,B23,B24)

Figure 9.3: *The DB and DDB functions*

Errors to avoid

The **DB** and **DDB** functions are fairly straightforward, but be aware that month is expressed as a number from 1 to 12 corresponding to a particular number rather than a text value (for example, "July") of that month.

DISC function

The **DISC** function calculates the "discount rate" being applied to a security—in other words, it calculates the rate of return on an investment in a non-coupon-paying security based on the current price and a given redemption value.

Uses and applications

DISC essentially calculates the rate of return expected on a zero coupon bond or other investment where there are no payments being made before redemption. While the same calculation could be approximated with the **RATE** function, the advantage of **DISC** is that it can easily account for very precise settlement and maturity dates.

The formula

=DISC(Settlement date, maturity date, price, redemption value, basis)

The Basis is a day count convention that is defined as follows:

0—30/360

1—Actual/Actual

2—Actual/360

3—Actual/365

4—European 3/360

Figure 9.4 shows the **DISC** function in action. As the example shows, compared with **RATE**, where a set number of periods is used, **DISC** enables the user to pick a settlement/purchase date and a specific maturity date, and then based on the price and the value at maturity (redemption value or par value), the function calculates the effective discount rate.

30	Settlement Date	10/15/2023
31	Maturity Date	1/25/2025
32	Price	92
33	Redemption Value	100
34	Basis	1
35		
36	6.25%	=DISC(B30,B31,B32,B33,B34)

Figure 9.4: *The DISC function*

Errors to avoid

The DISC function is easy to use, but remember that if payments are being made to the holder of the investment, the discount rate will not be accurate because it does not take those payments into account.

DURATION and MDURATION functions

Duration and Modified Duration are two metrics that can be used to estimate the impact on a bond's price from a change in interest rates. Excel calculates both with the **DURATION** and **MDURATION** functions.

Uses and applications

Calculating duration is an extremely common approach to estimating interest rate risk in bonds. While the calculation can be done manually, Excel offers a fast and easy alternative with the **DURATION** and **MDURATION** functions. **DURATION** and **MDURATION** are both measures of risk in bond portfolios, but they are different measures—**DURATION** is the Macauley duration which is the weighted average term to maturity of the cash flows from the security. In contrast, **MDURATION** is the modified duration of security, which is a more precise measure of the price sensitivity of a bond to changes in interest rates. The units on Macauley Duration are years, whereas the units on Modified Duration are percentages. In practice, while the two measures are different numbers, they are mathematically linked together, and both provide reasonable measures for risk in a fixed-income portfolio.

The formula

=DURATION(settlement date, maturity date, coupon, yield, frequency, basis)

=MDURATION(settlement date, maturity date, coupon, yield, frequency, basis)

Figure 9.5 illustrates the use of the **DURATION** function with a few of the possible characteristics shown:

39	Settlement Date	10/15/2023		
40	Maturity Date	1/24/2040		
41	Coupon	5%		
42	Yield	7%		
43	Frequency	2		
44	Basis	1		
45				
46	10.54529853	=DURATION(B39,B40,B41,B42,B43,B44)		
47	10.18869423	=MDURATION(B39,B40,B41,B42,B43,B44)		

Figure 9.5: *The DURATION and MDURATION functions*

Errors to avoid

The inputs for both duration options are identical and mostly straightforward. Be careful to express coupon as a percent of par rather than a dollar amount, though.

EFFECT function

The **EFFECT** function calculates the annual effective interest rate given a nominal rate and a number of compounding periods.

Uses and applications

The **EFFECT** function is a quick tool for determining the effective rate of interest, starting from a nominal rate and a given number of compounding periods. This is useful when dealing with any interest rate that is stated in nominal terms but compounded more frequently, such as with credit cards.

The formula

=EFFECT(nominal rate, number of compounding periods)

Figure 9.6 illustrates how to use the **EFFECT** function:

50	Nominal Rate	12%
51	Number of Periods	12
52	12.68%	=EFFECT(B50,B51)

Figure 9.6: *The EFFECT function*

Errors to avoid

The **EFFECT** function is easy to use, and you are unlikely to make any mistakes with it as long as you understand what the inputs are.

Time Value of money functions

Excel has a set of five-time value of money functions that can be used for myriad finance problems: **PV**, **FV**, **NPER**, **RATE**, and **PMT.**

Uses and applications

The time value of money is incredibly important in many aspects of finance and investing. Traditionally, these calculations were done with a financial calculator

or sometimes by hand, but Excel's functions provide an easy alternative method for finding **present value** (**PV**), **future value** (**FV**), an **annuity payment** (**PMT**), a yield or **rate** of interest on a series of cash flows (**RATE**), or the **number of periods required in a financial problem** (**NPER**). It is hard to overstate the importance of the time value of money in finance and business, so these functions are very useful for anyone in the business world who needs to do analytical calculations or make investment decisions.

The formula

=FV(Rate, Number of Periods, Payment, Present Value, payment timing indicator)

=PV(Rate, Number of Periods, Payment, Future Value, payment timing indicator)

=RATE(Number of Periods, Payment, Present Value, Future Value, payment timing indicator)

=NPER(Rate, Payment, Present Value, Future Value, payment timing indicator)

=PMT(Rate, Number of Periods, Present Value, Future Value, payment timing indicator)

Any of the five-time value of money inputs can be solved for using the appropriate formula if the other four variables are known. For instance, future value can be solved for with present value, payment, interest rate, and the number of periods. Similarly, payment can be solved for with present value, future value, interest rate, and the number of periods. Each of the formulas also has a fifth input—an indicator variable for the timing of cash flows—at the beginning or end of each period (also called an annuity due or an ordinary annuity).

Figure 9.7 shows all five-time values of money functions. As the figure illustrates, given any four of the five variables, the remaining variable can be solved for:

54	Present Value	1079
55	Future Value	-4000
56	Payment	200
57	Rate	10%
58	Number of Periods	7
59	Ordinary Annuity?	0
60	($4,000.10)	=FV(B57,B58,B56,B54,B59)
61	$1,078.95	=PV(B57,B58,B56,B55,B59)
62	10%	=RATE(B58,B56,B54,B55,B59)
63	6.99982522	=NPER(B57,B56,B54,B55,B59)
64	$199.99	=PMT(B57,B58,B54,B55,B59)

Figure 9.7: *The PV, FV, RATE, PMT, and NPER functions*

Errors to avoid

Time value of money issues can be complex, and a fuller discussion of them is really beyond the scope of this book. Outside of conceptual errors with these functions, such as calculating future value when you should be calculating present value, the most common errors in actually using the functions are likely to be around what order to put the arguments into the Excel functions. Be careful to make sure you are using the correct value for each argument and not, for example, putting the rate value in for the number of periods.

INTRATE function

The **INTRATE** function gives the interest rate that an investor earns on security, where the invested amount is held to maturity.

Uses and applications

INTRATE is similar to **DISC**—both of them produce percentage rates for an investment based on the same set of inputs. The major difference is that **DISC** is the discount rate and essentially determines how much the future value is discounted each year to get to the initial price or starting value. In contrast, **INTRATE** determines the rate of return starting from the initial investment or price to get to the future redemption value. As a result, **INTRATE** is usually a larger number than **DISC**.

The formula

=INTRATE(Settlement date, Maturity date, Price, Redemption value, Basis)

The inputs for **INTRATE** are the same as for **DISC** and a fairly self-explanatory. The final argument—Basis—is optional and is an indicator variable related to the days per year convention being used in the calculation. *Figure 9.8* shows how **INTRATE** is used and how the output compares to **DISC**.

66			
67	Settlement Date	1/25/2023	
68	Maturity Date	1/25/2025	
69	Price	80	
70	Redemption Value	100	
71	Basis	1	
72			
73	12.49%	=INTRATE(B67,B68,B69,B70,B71)	
74	10.00%	=DISC(B67,B68,B69,B70,B71)	

Figure 9.8: *The INTRATE function*

Errors to avoid

Be sure that you are applying the correct function in Excel for your needs—**DISC** should be used to calculate a discount rate (most often in the context of an NPV calculation), whereas **INTRATE** gives the fully invested interest rate on an investment held to maturity.

IPMT function

The IPMT function is a handy tool for determining the amount of interest that will be paid as part of an amortized loan payment in any given period.

Uses and applications

IPMT is similar to **CUMIPMT** in that both calculate the interest portion of an amortized loan payment. However, **CUMIPMT** calculates the cumulative amount of interest that will be paid on the loan, whereas IPMT simply calculates the interest amount in that specific period. IPMT also enables the user to set the future balance due on the loan at the end of the term rather than simply assuming a complete paydown. However, because amortized loan payments contain both principal and interest, the amount of the loan going toward interest falls over the life of the loan, as shown in *figure 9.9*.

The formula

=IPMT(Interest rate, the period when interest amount should be calculated, number of periods in the loan, present value, future value, payment timing – beginning or end of the period)

Figure 9.8 illustrates the use of the **IPMT** function under two scenarios—one for the interest portion of the loan payment in the first period (first payment) of the loan and the other for the tenth period or payment of the loan (when the outstanding principal has been partially paid down).

78	Period	1		Period	10		
79	Rate	5.20%		Rate	5.20%		
80	Number of Periods	36		Number of Periods	36		
81	PV	1000		PV	1000		
82	FV	0		FV	0		
83	($52.00)	=IPMT(B79,B78,B80,B81,B82,0)		($46.22)	=IPMT(E79,E78,E80,E81,E82,0)		

Figure 9.9: *The IPMT function*

Errors to avoid

The **IPMT** is a fairly straightforward formula—just be sure you understand what it is calculating and the loan period that you are looking for.

IRR, MIRR, and NPV functions

Just as Excel has a set of functions related to the time value of money, it also has a set of functions that can be used for capital budgeting. The most important of these functions are IRR, MIRR, and NPV.

Uses and applications

Capital budgeting is the decision-making process used in finance to evaluate large investment opportunities. Three of the primary capital budgeting criteria are internal rate of return, modified internal rate of return, and net present value, and Excel has a function for each calculation—**IRR**, **MIRR**, and **NPV**, respectively.

The formula

=IRR(Cash Flows, guess IRR)

=MIRR(Cash Flows, Financing Rate, Reinvestment Rate)

=NPV(Discount rate, Cash Flows)

Capital budging is a major area in corporate finance, and so **IRR**, **MIRR**, and **NPV** are all widely used functions in business. All three require a set of projected cash flows from an investment, like the ones shown for Years 1–5 in the example illustrated in *figure 9.9*. From there, IRR also uses a "guess" value as its second argument—this is a (optional) decimal value that represents the estimated percentage the user thinks will be close to the IRR. This argument is optional, and if not provided, Excel uses 0.10, corresponding to 10%. The reason this "guess" is used is because Excel essentially solves for the IRR by trial and error, and the guess gives the software a starting point to begin at.

IRR is widely used as a capital budgeting measure, but it is limited by the inability to account for reinvestment rates on cash flows. **MIRR** solves this issue by adding two additional arguments beyond the cash flows—the financing rate representing the cost of capital and the reinvestment rate representing the rate of return that can be earned on cash flows paid out after the project begins. Both **MIRR** and **IRR** produce percentages that essentially capture the rate of return on the investment.

The **NPV** function calculates the net present value of a series of cash flows. It simply requires the set of future cash flows and the discount rate to be applied to them, and

then, it will produce a dollar figure that represents the wealth or "profit" generated by the investment in question. However, the **NPV** function implicitly assumes that the first cash flow occurs one year in the future—meaning that if an upfront investment is required, that cash flow must be excluded from the **NPV** calculation and then added to the end of the equation, as shown in *figure 9.9*. All three of the capital budgeting functions also assume equal time intervals between the cash flows, so if that is not an accurate assumption, zero values may be added to create a uniform series of cash flows over time (for example, if the cash flow is 1,000 in Year 1 and 3,000 in Year 4, adding 0 for Years 2 and 3 will fix the problem, resulting in Excel showing 1,000, 0, 0, 3,000 for Years 1 to 4).

Figure 9.10 illustrates the use of the **IRR**, **MIRR**, and **NPV** functions:

85	Year	Cash Flow		
86	0	-5000		
87	1	1000		
88	2	1500		
89	3	3000		
90	4	5000		
91	5	6500		
92	Financing Rate	12%		
93	Reinvestment Rate	15%		
94				
95	IRR	41%	=IRR(B86:B91)	
96	MIRR	32%	=MIRR(B86:B91,B92,B93)	
97	NPV	$6,089.85	=NPV(B92,B87:B91)+B86	

Figure 9.10: *The capital budgeting functions IRR, MIRR, and NPV*

Errors to avoid

Be careful to add any upfront "year 0" investments to the end of the **NPV** formula rather than including them directly in the function—this is a very common mistake that results from Excel implicitly assuming that **NPV** cash flows begin in Year 1 rather than Year 0.

NOMINAL function

The **NOMINAL** function is a simple function that calculates the nominal interest rate given an effective rate and a number of compounding periods.

Uses and applications

Nominal rates are typically quoted interest rates for a one-year period, but in some cases, interest may be compounded more frequently, resulting in an effective rate

that is different than the nominal rate. The **EFFECT** function allows an Excel user to calculate the effective rate given the nominal yield and a number of compounding periods, whereas **NOMINAL** goes the other way and calculates the nominal rate based on the effective rate and the number of compounding periods.

The formula

=NOMINAL(Effective rate, number of compounding periods)

The **NOMINAL** formula is easy to use, as illustrated by *figure 9.10*:

100	Effective Rate	12.68%
101	Number of Periods	12
102	12.00%	=NOMINAL(B100,B101)

Figure 9.11: *The NOMINAL function*

Errors to avoid

The **NOMINAL** function is straightforward to use—check your cell references to avoid simple errors, but other than that, you are unlikely to have any issues.

PRICE, YIELD, YIELDDISC, and YIELDMAT functions

Excel has a series of functions related to calculating bond prices and yields. The most notable of these are **PRICE** (for calculating a bond's price), **YIELD** (for calculating a bond's yield), **YIELDDISC** (for calculating the yield on a zero-coupon discounted security like a Treasury bill), and **YIELDMAT** (for calculating the yield to maturity on a bond).

Uses and applications

Excel is widely used in investment circles for price calculations, and it is very handy, especially for bonds. The price and yield functions enable quick calculations of what a bond's fair value (or fair yield) is under a variety of scenarios.

The formula

=PRICE(Settlement date, Maturity date, Coupon rate, Yield, Redemption or Par Value, Frequency of coupon payments per year, Basis)

=YIELD(Settlement date, Maturity date, Coupon rate, Price, Redemption or Par Value, Frequency of coupon payments per year, Basis)

=YIELDDISC(Settlement date, Maturity date, Price, Redemption or Par Value, Basis)

=YIELDMAT(Settlement date, Maturity date, Issuance Date, Coupon Rate, Price, Basis)

Figure 9.12 illustrates the four functions in question. The **PRICE** and **YIELD** functions are essentially opposites of one another—given the settlement date, maturity date, coupon rate, pay value, frequency, and Basis, one can either find the price (with the securities' yield) or yield (with the bond's price). These two functions are the more useful ones in most situations. As in other functions, the Basis is an indicator variable related to the days per year convention being used. It is an optional argument with a default value of 0 (actual/actual). The other arguments in each function are fairly straightforward, though be sure to include the coupon as a rate rather than a dollar amount.

The **YIELDDISC** and **YIELDMAT** functions are more specialized—**YIELDDISC** is used when there is a discounted bond that does not pay a coupon that needs to be priced. **YIELDMAT** gives the yield to maturity for a bond from the issue date to the maturity date.

Refer to *figure 9.12:*

104	Settlement Date	1/25/2023		
105	Maturity Date	1/25/2025		
106	Issue Date	1/15/2020		
107	Coupon	8.50%		
108	Yield	9.50%		
109	Price	98		
110	Redemption Value	100		
111	Frequency	2		
112	Basis	0		
113				
114	98.22	=PRICE(B104,B105,B107,B108,B110,B111,B112)		
115	9.62%	=YIELD(B104,B105,B107,B109,B110,B111,B112)		
116	1.02%	=YIELDDISC(B104,B105,B109,B110,B112)		
117	7.68%	=YIELDMAT(B104,B105,B106,B107,B109,B112)		

Figure 9.12: *The PRICE, YIELD, YIELDDISC, and YIELDMAT functions*

Errors to avoid

Each of the four functions discussed here serves a different purpose—it is sometimes tempting to try and make one function like Yield work for other scenarios like a zero

coupon bond pricing exercise—doing that risks making an error because of how Excel calculates each function. Stick to using the functions strictly for their intended purpose.

XIRR and XNPV functions

The XIRR and XNPV functions offer more powerful capital budgeting tools for dealing with specific cash flows expected at specific dates.

Uses and applications

While **IRR** and **NPV** are the default standard tools for capital budgeting, one limitation of them in Excel has always been that it is difficult to do calculations where cash flows are spread out over uneven periods of time. For instance, a project might have a series of cash flows every three months for the first two years and then annually on different dates thereafter. In this case, either a complicated series of cash flows with many zero values had to be built, or else **IRR** and NPV would only provide approximations of the true value. **XIRR** and **XNPV** correct this problem by allowing the user to specify on what specific date each cash flow will occur.

The formula

=XIRR(Cash flow amounts, Cash Flow Dates)

=XNPV(Financing rate, Cash flow amounts, Cash Flow Dates)

Figure 9.13 illustrates **XIRR** and **XNPV**. Notice that the dates in the year column vary greatly and are not spread uniformly over time. In fact, any series of dates can be used with **XNPV** and XIRR, so the bigger issue becomes how confident we are about what specific date a cash flow will be received. In some cases, like with a loan payment, we may know with precision when cash flows will be received, and in other cases, like a business investment, we may just be guessing. If we know with precision when the payments will occur, **XIRR** and **XNPV** are preferable to **IRR** and **NPV**.

120	Year	Cash Flow	
121	1/1/2024	-5000	
122	4/15/2025	1000	
123	6/12/2026	1500	
124	11/1/2027	3000	
125	9/1/2028	5000	
126	6/1/2029	6500	
127	Financing Rate	12%	
128			
129	IRR	34%	=XIRR(B121:B126,A121:A126)
130	NPV	$5,405	=XNPV(B127,B121:B126,A121:A126)

Figure 9.13: *The XIRR and XNPV functions*

Errors to avoid

Be careful to specify date values for **XIRR** and **XNPV**—text values such as "July 1, 2025" will not be ready correctly by Excel as a date and will trigger an error.

VDB function

The **VDB** function offers a more powerful depreciation calculation function than **DB** and **DDB**.

Uses and applications

The **VDB** function enables Excel users to vary the speed of depreciation of an asset based on a value the user sets. It also has an argument that switches the depreciation to a straight line in the event that the depreciation in a particular period is smaller than what would result from the straight line method.

The formula

=VDB(Cost, Salvage value, Asset life, Starting period for depreciation, Ending period for depreciation, Depreciation factor, Switch to Straight Line method indicator)

The **VDB** function essentially lets the user customize the deprecation calculation more than they can with **DB** or **DDB**. While Cost, Salvage value, and Asset life all remain standard, **VDB** enables the user to look at deprecation for particular start and end periods and to change the speed or factor used in deprecation. The default factor is 2, corresponding to the double declining balance method, but other values such as 1, 1.5, or 3 can also be used if the situation calls for it. The final argument in the function is a switch indicator variable that will cause Excel to switch to using Straight Line depreciation if the depreciation amount in a given year is lower than what the Straight Line method would produce.

Figure 9.14 illustrates the use of the **VDB** function:

133	Cost	5000		
134	Salvage	300		
135	Life	5		
136	Start Period	2		
137	End Period	3		
138	Depreciation Factor	2		
139	Switch?	FALSE		
140				
141	$720.00	=VDB(B133,B134,B135,B136,B137,B138,B139)		

Figure 9.14: *The VDB function*

Errors to avoid

The **VDB** function is relatively simple to use—just be sure you understand the inputs, and you are unlikely to make any mistakes.

Conclusion

The functions shown in this chapter are financial functions, which you are likely to use repeatedly in any work in the business world. The capital budgeting, time value of money, and depreciation functions are critical for accountants, financial professionals, and investors to be able to use, so commit these to memory and make sure you understand the inputs that go into the functions.

Multiple choice questions

1. **You have been asked to examine the effect of rising interest rates on a portfolio of bonds—which function can help you with this task?**
   a. XIRR
   b. DURATION
   c. DISC
   d. WEEKDAY

2. **If you wanted to determine the investment return as a percentage from a series of cash flows resulting from an upfront investment in a new business venture, what function could you use?**
   a. NPV
   b. IRR
   c. DURATION
   d. VALUE

3. **If you wanted to determine the current value of a payment that will be received in the future based on a given discount rate, what function is most appropriate?**
   a. =PMT
   b. =RATE
   c. =FV
   d. =PV

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 10

Date and Time Functions in Excel

Introduction

Dates and times are a persistent hassle to deal with in datasets—they look like regular numbers—but they are not. As a result, it is important to be able to use special Excel functions that let us work with dates and times when building larger models and datasets. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- DATE function
- DAY, MONTH, and YEAR function
- DAYS and DAYS360 function
- EDATE and EOMONTH function
- HOUR, MINUTE, and SECOND functions
- ISOWEEKNUM and WEEKNUM functions
- NETWORKDAYS function

- NOW and TODAY functions
- TIME function
- WEEKDAY function

Objectives

After studying this unit, you will be able to use a variety of date and time functions in Excel.

DATE function

The **DATE** function is a handy tool to turn separate months, days, and years into cohesive dates.

Uses and applications

In many datasets, months, days, and years may be stored in separate variables (for example, columns). Getting a date out of these separate values can be tedious, but the DATE function makes it easy.

The formula

=DATE(Year, Month, Day)

Figure 10.1 illustrates the use of the **DATE** function:

	A	B	C	D	E	F
1	Date	Year	Month	Day		
2	1/5/2023	2023	1	5	=DATE(B2,C2,D2)	
3	4/7/2024	2024	4	7	=DATE(B3,C3,D3)	
4	6/19/2025	2025	6	19	=DATE(B4,C4,D4)	
5	11/26/2026	2026	11	26	=DATE(B5,C5,D5)	

Figure 10.1: *The DATE function*

When using this function, we may get a number (as a result of the formula), such as 41230, instead of a date. Generally, this occurs due to the formatting of the cell. The function returned the correct value, but the cell is displaying the date serial number instead of the formatted date.

Hence, we need to change the formatting of the cell to display a date. The easiest and quickest way to do this is to select the cell(s) to be formatted and then select the

Date cell formatting option from the drop-down menu in the "Number" group on the Home tab (of the Excel ribbon)

Errors to avoid

The **DATE** function is simple to use, but be sure that you do not mix up the ordering of the arguments in the function—it goes from largest unit to smallest—year, month, day.

DAY, MONTH, and YEAR functions

Just as the **DATE** function allows the Excel user to convert from separate days, months, and years to a single date, the **DAY**, **MONTH**, and **YEAR** functions allow the Excel user to do the opposite and pull out the specific day, month, and year from a single date.

Uses and applications

In many applications, we may need a day, month, or year from a specific date—for instance, in trying to identify seasonality in a business' sales, we probably do not care about the specific date that a sale occurred, but we do care about the month that the sale occurred in. When dealing with issues like this, the **DAY**, **MONTH**, and **YEAR** functions can provide quick ways to pull specific values from a date itself.

The formula

=DAY(Date value)

= MONTH(Date value)

=YEAR(Date value)

Figure 10.2 demonstrates how to use the **DAY**, **MONTH**, and **YEAR** functions:

8	Day	Month	Year	Date
9	1	1	2024	1/1/2024
10	13	7	2024	7/13/2024
11	28	12	2025	12/28/2025
12	=DAY($D9)	=MONTH($D9)	=YEAR($D9)	

Figure 10.2: *The DAY, MONTH, and YEAR functions*

Errors to avoid

The **DAY**, **MONTH**, and **YEAR** functions are easy to use—just be sure that they are pulling from an actual date and not a text value such as "January 5 2025." This will generate an error with the function.

DAYS and DAYS360 functions

In many situations, we may need to know the number of days between two different dates—and that is exactly what the **DAYS** and **DAYS360** functions do.

Uses and applications

When calculating changes over time (for example, sales growth, how long an employee worked for a firm), rates of interest, or a variety of other business metrics, we often need to know the number of days between two dates. That is an easy calculation for short periods of time, but a much harder one for longer periods, especially across several years. The **DAYS** and **DAYS360** functions simplify this process. **DAYS** calculates the days between two dates, while **DAYS360** calculates the days between dates based on the 360 days per year simplifying assumption, which is often used in finance.

The formula

=DAYS(Ending Date, Starting Date)

=DAYS360(Starting Date, Ending Date, optional 0 or 1 indicator for the US or European calculation method)

Figure 10.3 illustrates the use of the **DAYS** and **DAYS360** functions. The default for the third argument is the US calculation method, but in practice, the two methods are almost identical, so this argument rarely has an impact on the output.

14	Days			Date
15	194	=DAYS(D16,D15)		1/1/2024
16	192	=DAYS360(D15,D16,0)		7/13/2024

Figure 10.3: *The DAYS and DAYS360 functions*

Errors to avoid

Note that **DAYS** and **DAYS360** have arguments in the opposite order—for DAYS, we start with the ending date and then add the beginning date as the second argument. On the other hand, for **DAYS360**, it is the opposite.

EDATE and EOMONTH function

EDATE and **EOMONTH** are simple functions that tell the user what a future date is based on a starting date and the number of months into the future.

Uses and applications

In some applications, such as accounts receivable or payable, student admissions to universities, or real estate leases, we may need to calculate a future date based on a starting date and an agreed-upon term. For instance, a customer might buy a product from a business and then have three months to pay. Calculating the date that a payment is due, in this scenario, is very easy, but calculating it for many customers, and recording those dates as new customers come in, is much more time-consuming. **EOMONTH** (essentially "ending of month") and **EDATE** (essentially "ending date") make this task easy.

The formula

=EDATE(starting date, number of months in the future)

=EOMONTH(starting date, number of months in the future)

The arguments for **EDATE** and **EOMONTH** are the same—the difference between the two functions is that **EDATE** finds a specific date that falls a given number of months in the future, whereas **EOMONTH** finds the last date in a particular month that occurs a given number of months in the future. *Figure 10.4* illustrates both functions:

	A	B	C	D
19				
20	4/30/2024	=EOMONTH(D20,3)		1/20/2024
21	7/20/2024	=EDATE(D20,6)		

Figure 10.4: *The EDATE and EOMONTH functions*

Errors to avoid

EDATE and **EOMONTH** are both simple functions, so just be sure you understand the difference between them and use the correct one for your application.

HOUR, MINUTE, and SECOND functions

Just as **YEAR**, **MONTH**, and **DAY** allow an Excel user to break down a date into component parts, **HOUR**, **MINUTE**, and **SECOND** does the same thing with times during the day.

Uses and applications

In some cases, when doing work in Excel, it makes sense to break times of the day down into hours, minutes, or even seconds. For instance, if we are trying to schedule

workers for particular shifts in a store, it is helpful to know the number of customers that come through during a particular hour of the day. We can determine this using the **HOUR** function if we have the time of day when a customer walks into a store.

The formula

=HOUR(time)

=MINUTE(time)

=SECOND(time)

Figure 10.5 shows how to use the **HOUR**, **MINUTE**, and **SECOND** functions.

	A	B	C	D
22				
23				
24	Time	Hour	Minute	Second
25	8:53:41	8	53	41
26	13:11:24	13	11	24
27	1:36:59 PM	13	36	59
28	2:30 PM	14	30	0
29		=HOUR(A26)	=MINUTE($A26)	=SECOND($A26)

Figure 10.5: *The HOUR, MINUTE, and SECOND functions*

Errors to avoid

The **HOUR**, **MINUTE**, and **SECOND** functions are simple to use, but be sure you are dealing with defined times in Excel that are formatted as time values. Spacing matters a lot for Excel formatting—for instance, note that Excel will not read "2:30 PM" as a time, but it will read "2:30 PM" as a time.

ISOWEEKNUM and WEEKNUM functions

The **ISOWEEKNUM** and **WEEKNUM** functions tell us the week number of a particular date in a year. The difference between the two functions is that **ISOWEEKNUM** uses the week containing the first Thursday of the year as Week 1, whereas **WEEKNUM** uses the week containing January 1 as Week 1.

Uses and applications

In some applications, we may need to know the number of weeks that have elapsed in a year when looking at a specific date. Let us consider, for example, that we are billing a customer based on an annual subscription, but they are signing up partway into the year. In this type of situation, **WEEKNUM** and **ISOWEEKNUM** (which follow the ISO 8601 date and time measurement standard) are very helpful functions.

The formula

=ISOWEEKNUM(date)

=WEEKNUM(date)

Figure 10.6 illustrates the use of the **WEEKNUM** and **ISOWEEKNUM** functions. You will notice that the functions produce identical results for two of the dates but different results for the third date.

	A	B	C	D
31	Week Number			
32	1	=ISOWEEKNUM(D32)		1/1/2024
33	47	=ISOWEEKNUM(D33)		11/21/2029
34	28	=ISOWEEKNUM(D34)		7/13/2014
35	1	=WEEKNUM(D35)		1/1/2024
36	47	=WEEKNUM(D36)		11/21/2029
37	29	=WEEKNUM(D37)		7/13/2014

Figure 10.6: *The ISOWEEKNUM and WEEKNUM functions*

Errors to avoid

Be sure you understand the difference between **WEEKNUM** and **ISOWEEKNUM** and are applying the correct function based on the approach to measuring the weeks that you want to use.

NETWORKDAYS function

The **NETWORKDAYS** function finds the number of working (that is, week) days that occur between two dates.

Uses and applications

NETWORKDAYS differs from **DAYS** because it examines only weekdays, where **DAYS** include all days. Thus, **NETWORKDAYS** is more useful in some applications, such

as calculating accrued vacation time for a human resources department, where presumably, we are only interested in working days rather than all days that have occurred. **NETWORKDAYS** also has an optional third argument that enables us to exclude any holidays from the calculation in the same way that weekends are excluded.

The formula

=NETWORKDAYS(Beginning date, ending date, any holidays that should be excluded from the calculation).

The final argument, holiday dates, is optional and can be ignored if there are no relevant holidays to include. *Figure 10.7* illustrates the use of the **NETWORKDAYS** function:

40	231	=NETWORKDAYS(D40,D41,D42:D44)		1/1/2024
41				11/21/2024
42				2/20/2024
43				5/29/2024
44				7/4/2024

Figure 10.7: *The NETWORKDAYS function*

Errors to avoid

The **NETWORKDAYS** and **DAYS** both calculate the number of days between two dates—but **NETWORKDAYS** excludes weekends. Be sure you are using the function that is right for your application.

NOW and TODAY functions

NOW and **TODAY** both serve similar purposes—they produce the current date that the formula calculated (**TODAY**) and the current date and time that the formula calculated (**NOW**).

Uses and applications

In finance applications, we often need to calculate the amount of interest on a loan that is currently due. Rather than continually updating the current date (or date and time) each time the Excel sheet is opened, we can use the **TODAY** or **NOW** functions.

The formula

=TODAY()

=NOW()

Figure 10.8 demonstrates the use of the **TODAY** and **NOW** functions.

47	1/17/2023 13:13	=NOW()
48	1/17/2023	=TODAY()

Figure 10.8: *The TODAY and NOW functions*

Errors to avoid

Neither the **TODAY** nor the **NOW** functions have any arguments, and both are aptly named—it would be difficult to make a mistake while using either one by itself.

TIME function

The **TIME** function constructs a time of day from a separate hour, minute, and day.

Uses and applications

TIME can be used to create a time of day if we have specific values for hours, minutes, and seconds. Note, however, that hours must be in a 24-hour/military time format. For instance, 17 represents 5 hours after noon, rather than allowing for 5 to be used for both AM and PM.

The formula

=TIME(Hour, Minute, Second)

Figure 10.9 demonstrates how to use the **TIME** function:

51	Time	Hour	Minutes	Seconds	
52	11:01:05 AM	11	1	5	=TIME(B52,C52,D52)
53	3:04:07 AM	3	4	7	=TIME(B53,C53,D53)
54	5:06:19 PM	17	6	19	=TIME(B54,C54,D54)
55	6:11:26 AM	6	11	26	=TIME(B55,C55,D55)

Figure 10.9: *The TIME function*

Errors to avoid

The TIME function is fairly basic, but be sure you understand that hours must be on a 24-hour clock rather than a 12-hour clock. For instance, Excel will read hour 11 as 11 AM rather than 11 PM.

WEEKDAY function

The **WEEKDAY** function is a simple but handy function that identifies the day of the week represented by a particular date.

Uses and applications

When dealing with dates, it can be a nuisance to determine what day of the week a particular date represents. For instance, 1/1/2024 could be a Wednesday, a Monday, or any other day potentially. Yet, we may need to know that information when determining schedules or for other planning purposes. **WEEKDAY** solves this problem by identifying the day of the week of a specific date, with Monday being 1 and Sunday being 7.

The formula

=WEEKDAY(date)

Figure 10.10 illustrates the use of the **WEEKDAY** function. As the figure illustrates, the output from the function is a number from 1 to 7 rather than a named day (for example, 1 instead of Monday).

59	Day Number		
60	2	=WEEKDAY(C60)	1/1/2024
61	4	=WEEKDAY(C61)	11/21/2029
62	1	=WEEKDAY(C62)	7/13/2014

Figure 10.10: *The WEEKDAY function*

Errors to avoid

Be sure you understand what the numbers mean in the output—1 is Monday, 2 is Tuesday, and so on. In some calendars and regions, the "first" day of the week can be Sunday or another day by convention, but for Excel output, 1 means Monday.

Conclusion

The functions shown in this chapter are date and time functions that can be very helpful for creating schedules or building models that rely on dates and times. The functions here are fairly simple to use, but a key issue is to be sure that the format of the inputs or outputs is the desired type. For instance, Excel will often output long serial numbers representing dates, but these can be easily changed to a date format in Excel to make them more readable.

Points to remember

- **`DATE`** function: Constructs a date from a year, month, and day.
- **`DAYS`** function: Calculates the number of days between two dates.
- **`EDATE`** function: Determines a specific date, a given number of months in the future, based on a starting date.
- **`HOUR`** function: Outputs the hour of the day from a particular time.
- **`MONTH`** function: Produces the month of the year given a date.
- **`NETWORKDAYS`** function: Calculates the number of working days between two dates.
- **`TODAY`** function: Outputs the current date that the formula is calculated.
- **`WEEKDAY`** function: Produces the day of the week for a given date.
- **`WEEKNUM`** function: Produces the week number in a year for a given date.
- **`YEAR`** function: Determines the year for a particular date.

Multiple choice questions

1. **What function is intended can be used to calculate the number of weekdays between two dates?**
    a. EDATE
    b. NETWORKINGDAYS
    c. DAYS
    d. WEEKDAY

2. **What function would produce the day of the week for a given date?**
    a. DAYS360
    b. WEEKDAY
    c. TODAY
    d. TIME

3. **If you wanted to build a variable representing time from a series of values giving hours, minutes, and seconds, what function could be used?**
   a. =DAY
   b. =NOW
   c. =HOUR
   d. =TIME

Answers

1. **b**
2. **b**
3. **d**

Chapter 11
Database Functions in Excel

Introduction

Excel is not the most powerful database software on the market—but it is widely used for database work because it is ubiquitous, simple to learn, and performs well for a variety of small database tasks. Moreover, in this role, it is important to understand the database functions that Excel includes. In this chapter, we will cover the most popular and useful formulas that you need to know in order to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- DAVERAGE function
- DCOUNT function
- DCOUNTA function
- DGET function
- DMIN and DMAX functions
- DPRODUCT function
- DSTDEV and DVAR functions

Objectives

After studying this unit, you will be able to use database functions for calculations in Excel.

DAVERAGE function

The **DAVERAGE** function enables the Excel user to find an average based on one or more criteria for a particular field in a database.

Uses and applications

Excel has a number of average functions—most notably **AVERAGE** and **AVERAGEIF**. These work fine for many applications, but for large databases where we may be looking for an average based on multiple criteria, **DAVERAGE** is often more effective. The **DAVERAGE** function lets us take an average across a particular field (column) based on certain criteria. This is especially so if we have multiple criteria that we want to segment the data by—**DAVERAGE** is generally more effective than **AVERAGEIF**.

The formula

=DAVERAGE(cells for the database, the field to be averaged, the cells containing the criteria)

The database functions are complex and finicky. The cells for the database need to contain the names of the fields in the columns with the values for each field in the rows. The top row referenced in the formula must contain the field names, and these field names have to be the same between the database and the criteria. The field to be averaged can be either the name embedded in quotations, as seen in *figure 11.1* or the number of columns for the field. The criteria have to be stored in a separate part of the sheet under the same field names as the database. Mathematical operators such as <, >, <> (not), <=, >=, and = can all be used with numerical criteria as illustrated.

Figure 11.1 illustrates the use of the **DAVERAGE** function:

	A	B	C	D	E	F	G
1	Individual	Exam	Score		**Criteria**		
2	A	1	73		Individual	Exam	Score
3	A	2	81		A		>80
4	A	3	90				
5	B	1	82		**Average**		
6	B	2	86		85.5		
7	B	3	84		=DAVERAGE(A1:C10,"Score",E2:G3)		
8	C	1	85				
9	C	2	67				
10	C	3	75				
11							

Figure 11.1: *The DAVERAGE function*

Errors to avoid

The **DAVERAGE** function is tricky to implement because of Excel's hidden rules about how the database and criteria have to be represented—try doing a simple example like the one shown here and make sure it gives you the output you expect before moving on to the actual calculations in a larger dataset.

DCOUNT function

DCOUNT is similar to **COUNTIF**—it enables the Excel user to count the number of rows that meet some specific criteria in a dataset.

Uses and applications

Just as **DAVERAGE** is often easier to use for multiple criteria than **AVERAGEIF**, **DCOUNT** is often a better choice than **COUNT** or **COUNTIF**.

The formula

=DCOUNT(cells for the database, the field to be counted, the cells containing the criteria)

Figure 11.2 demonstrates how to use the **DCOUNT** function:

	A	B	C	D	E	F	G
1	Individual	Exam	Score		**Criteria**		
2	A	1	73		Individual	Exam	Score
3	A	2	81		A	>1	
4	A	3	90				
5	B	1	82		**Average**		
6	B	2	86		2		
7	B	3	84		=DCOUNT(A1:C10,3,E2:G3)		
8	C	1	85				
9	C	2	67				
10	C	3	75				

Figure 11.2: *The DCOUNT function*

Errors to avoid

Remember—the field names in the database and the criteria must match **EXACTLY**, including spaces in the names, or else Excel will not produce correct results.

DCOUNTA function

The **DCOUNTA** function counts non-blank cells in a database that meet certain criteria.

Uses and applications

DCOUNT and **DCOUNTA** are very similar—the only difference is that **DCOUNTA** counts non-blank cells, whereas **DCOUNT** counts all cells that meet the criteria.

The formula

=DCOUNTA(cells for the database, the field to be counted, the cells containing the criteria)

Figure 11.3 shows the use of the **DCOUNTA** function:

	A	B	C	D	E	F	G
1	Individual	Exam	Score		**Criteria**		
2	A	1	73		Individual	Exam	Score
3	A	2	81		A	>1	
4	A	3					
5	B	1	82		**Average**		
6	B	2	86		1		
7	B	3	84		=DCOUNTA(A1:C10,3,E2:G3)		
8	C	1	85				
9	C	2	67				
10	C	3	75				

Figure 11.3: *The DAVERAGE function*

Errors to avoid

DCOUNTA has the same challenges as the other database functions—be careful about criteria and field names and how the database is structured.

DGET function

The **DGET** function pulls a single record from a database that matches a specified set of criteria.

Uses and applications

DGET is a unique function—the closest alternatives are probably **VLOOKUP** and **XLOOKUP**, but **DGET** is more flexible and easier to use with an established database. Essentially, the function pulls a single value that meets a set of criteria. If more than one value meets the criteria, the #NUM! error is returned, and if no values meet the criteria, the #VALUE! error is returned.

The formula

=DGET(cells for the database, the field to be pulled from, the cells containing the criteria)

Figure 11.4 illustrates the use of the **DGET** function:

	A	B	C	D	E	F	G
1	Individual	Exam	Score		**Criteria**		
2	A	1	73		Individual	Exam	Score
3	A	2	81		B	3	
4	A	3	90				
5	B	1	82		**Average**		
6	B	2	86		84		
7	B	3	84		=DGET(A1:C10,3,E2:G3)		
8	C	1	85				
9	C	2	67				
10	C	3	75				

Figure 11.4: *The DGET function*

Errors to avoid

The **DGET** function will return an error if either no records are found that meet the criteria (#VALUE!) or more than one record is found that meets the criteria (#NUM!).

DMIN and DMAX functions

The **DMIN** and **DMAX** functions find minimum and maximum values in a dataset based on specified criteria.

Uses and applications

DMIN and **DMAX** are unique Excel functions in that they offer the ability to pull a minimum or maximum value based on one or more specific criteria. This is very useful when examining databases of product sales, employee performance, student records, and so on.

The formula

=DMIN(cells for the database, the field to be averaged, the cells containing the criteria)

=DMAX(cells for the database, the field to be averaged, the cells containing the criteria)

Figure 11.1 illustrates the use of the **DMIN** and **DMAX** functions:

	A	B	C	D	E	F	G	H
1	Individual	Exam	Score		**Criteria**			
2	A	1	73		Individual	Exam	Score	
3	A	2	81		C			
4	A	3	90					
5	B	1	82		**Average**			
6	B	2	86		67	=DMIN(A1:C10,3,E2:G3)		
7	B	3	84		85	=DMAX(A1:C10,3,E2:G3)		
8	C	1	85					
9	C	2	67					
10	C	3	75					

Figure 11.5: *The DMIN and DMAX functions*

Errors to avoid

If you can use the other database functions and understand their idiosyncrasies, **DMIN** and **DMAX** provide no more challenges.

DPRODUCT function

The **DPRODUCT** function multiplies a series of values in the database by one another based on given criteria.

Uses and applications

DPRODUCT is less obviously useful than some of the other database functions, but it does have very specific invaluable niche uses. For example, when dealing with a database of products, we might need to know the volume they take up for packaging purposes, but we are more likely to have the height, length, and width of each product rather than the volume itself in the database. **DPRODUCT** enables us to calculate that volume based on those values.

The formula

=DPRODUCT(cells for the database, the field to be multiplied, the cells containing the criteria)

The database functions allow us to not only create specific criteria but also to specify **OR** type criteria. For instance, *figure 11.6* shows the use of **DPRODUCT**. The first **DPRODUCT** formula (cell E20) multiples the height, width, and length just for product A based on the criteria (3 × 4 × 5 = 60). However, the second **DPRODUCT** formula (cell E21, formula in F21) expands the criteria to allow either product A or B, resulting

in the values of product A being multiplied by product B (3 × 4 × 5 × 1 × 1 × 10 = 600). While that value is a bit nonsensical for the example given here of calculating product volumes, the formula illustrates an important point—if we want multiple **AND** criteria for a database function, we put them on the same row in Excel. If we want multiple **OR** criteria, we put them in multiple rows (with an expanded criteria field in the formula, of course).

Refer to *figure 11.6:*

	A	B	C	D	E	F	G
13							
14	Product	Dimension	Value		**Criteria**		
15	A	Height	3		Product	Dimension	Value
16	A	Width	4		A		
17	A	Length	5		B		
18	B	Height	1				
19	B	Width	1		**Volume**		
20	B	Length	10		60	=DPRODUCT(A14:C23,3,E15:G16)	
21	C	Height	85		600	=DPRODUCT(A14:C23,"Value",E15:G17)	
22	C	Width	67				
23	C	Length	75				

Figure 11.6: *The DPRODUCT function*

Errors to avoid

The "**AND**" versus "**OR**" criteria can be tricky at first—it is very useful since this works across all database functions. However, make sure you try it and understand it before applying it to complex Excel models.

DSUM function

The **DSUM** function enables the Excel user to find a sum based on one or more criteria for a particular field in a database.

Uses and applications

DSUM is similar to **SUMIF** but more powerful in that it enables the Excel user to sum up values that meet multiple criteria in a dataset (versus a single criterion with the basic SUMIF function).

The formula

=DSUM (cells for the database, the field to be summed, the cells containing the criteria)

DSUM and the other mathematical database functions are particularly powerful when combined with multiple AND/OR criteria. *Figure 11.7* adds up sales for Individual B in cell E32 (formula in cell F32). Then in cell E33, the criteria range is expanded to include either sale from individual B (82 + 86 + 84) or sales of 3 made by individual A (82 + 86 + 84 + 100).

Refer to *figure 11.7:*

	A	B	C	D	E	F	G
25							
26	Individual	Product	Sales		**Criteria**		
27	A	1	73		Individual	Product	Sales
28	A	2	81		B		
29	A	3	100		A	3	
30	B	1	82				
31	B	2	86		**Sum**		
32	B	3	84		252	=DSUM(A26:C35,3,E27:G28)	
33	C	1	85		352	=DSUM(A26:C35,3,E27:G29)	
34	C	2	67				
35	C	3	75				

Figure 11.7: *The DSUM function*

Errors to avoid

Remember that criteria on the same row, such as row 29 in the preceding figure are "and" criteria—sales from individual A and in product 3, whereas criteria on multiple rows are "or" criteria—like rows 28 and 29 covering individual B's sales or sales from individual A in product 3.

DSTDEV and DVAR functions

DSTDEV and **DVAR** offer standard deviation and variance calculations (respectively) for cells in a database that meet certain criterion.

Uses and applications

Excel has a sample (**DSTDEV**) and population (**DSTDEVP**) function for calculation standard deviations in a database based on various criteria. Similarly, for variance, it has a sample (**DVAR**) and population (**DVARP**) function. These functions are uniquely useful for finding standard deviations and variances in datasets under specific conditions—something that requires nested formulas (such as **IF** and **STDEV**) otherwise.

The Formula

=DSTDEV(cells for the database, the field to be averaged, the cells containing the criteria)

=DSTDEVP(cells for the database, the field to be averaged, the cells containing the criteria)

=DVAR(cells for the database, the field to be averaged, the cells containing the criteria)

=DVARP(cells for the database, the field to be averaged, the cells containing the criteria)

Figure 11.8 shows how to use these conditional standard deviation and variance formulas.

26	Individual	Product	Sales		**Criteria**		
27	A	1	73		Individual	Product	Sales
28	A	2	81			>1	
29	A	3	100				
30	B	1	82				
31	B	2	86		**Standard Deviation and Variance**		
32	B	3	84		11.13	=DSTDEV(A26:C35,3,E27:G28)	
33	C	1	85		10.16	=DSTDEVP(A26:C35,3,E27:G28)	
34	C	2	67		123.77	=DVAR(A26:C35,3,E27:G28)	
35	C	3	75		103.14	=DVARP(A26:C35,3,E27:G28)	

Figure 11.8: *The DVAR and DSTDEV functions*

Errors to avoid

Remember that there is a difference between a sample and a population standard deviation or variance. In essence, if you have all individuals in a dataset (like all students who have taken a test), you can use the population calculations. Otherwise, use the sample version of the functions.

Conclusion

The functions shown in this chapter are all for use with databases on values—they will let you go through and analyze values in the dataset or pull out specific pieces of information. These functions take a little getting used to initially and so they are not widely used by most Excel users, but they are very powerful and offer the ability to quickly do work in Excel that otherwise would require a complex formula.

Multiple choice questions

1. **What function is intended can be used to pull a specific value that meets given criteria in a dataset?**
    a. DSEARCH
    b. DGET
    c. DSUM
    d. DLOOKUP

2. **If you wanted to use a DSUM function to add up sales made by either salesperson A or B, how should the criteria be structured?**
    a. A and B on the same row in adjacent columns
    b. A and B in the same column on adjacent rows
    c. A and B in separate Excel worksheets
    d. There is no way to do this type of "or" criteria

3. **If you wanted to count the number of cells that met specific criteria and exclude all blank cells, which function would you use?**
    a. =DCOUNT
    b. =DSUM
    c. =DSUMA
    d. =DCOUNTA

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 12

Cube Functions in Excel

Introduction

Cube functions are an unusual group of functions with some specific niche uses. While you may not be familiar with them, they can be very important in specific settings. In this chapter, we will cover the most popular and useful formulas that you need to know in order to be able to use Excel effectively.

Essentially, Excel CUBE functions are like the new PivotTables. They allow us to build reports, based on Power Pivot models, in a more flexible fashion. PivotTables are very restrictive in their layout, and their cells cannot be edited. This is where the **CUBE** functions come in and solve these issues. Moreover, just like PivotTables, we can use Slicers to filter **CUBE** functions for fully interactive reports and dashboards.

Structure

In this chapter, we will discuss the following topics:

- CUBEMEMBER function
- CUBEKPIMEMBER function
- CUBEMEMBERPROPERTY function
- CUBERANKEDMEMBER function
- CUBEVALUE function

Objectives

After studying this unit, you will be able to use a variety of Cube functions in Excel.

CUBEMEMBER function

The **CUBEMEMBER** function returns a particular value from the data that is referenced in the function. It is a good basic **CUBE** function that can pull specific values from large data sets.

Uses and applications

To use the **CUBE** function, you need to put your data into Excel as a pivot table and then check the option **Add this data to the data model**, as shown in *figure 12.1*. From here, you can pull values from the data set using the **CUBE** function in the same way you would with a pivot table.

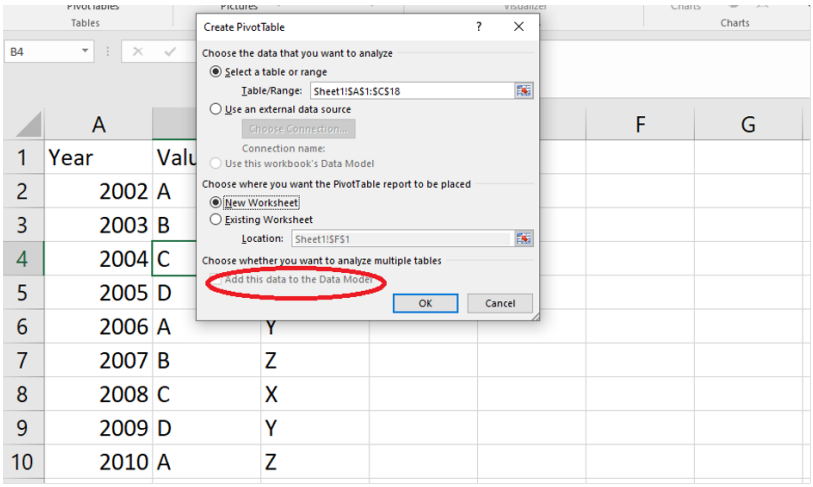


Figure 12.1: *Adding data for use with CUBE functions*

The formula

=CUBEMEMBER(connection, member-expression, optional caption)

Connection is the text string of the name of the connection to the cube.

Member_expression is the text string of a **multidimensional expression** (**MDX**) that refers to a unique member in the cube or a cell range.

Caption is an optional argument for any alternative text string that is displayed in Excel.

Errors to avoid

Note that **CUBEMEMBER** will only work if you have set up a data table for the function to pull from.

CUBEKPIMEMBER function

The **CUBEKPIMEMBER** function is a specialty function that only works when the Excel workbook is connected to a Microsoft SQL Server data source (generally 2005 and later). Essentially, the function pulls a specific value from the data source and returns it in Excel.

Uses and applications

CUBEKPIMEMBER is a function to pull a key performance indicator or KPI from an SQL server—the function enables Excel users to monitor specific KPIs based on data that is entered into the SQL database.

The formula

=CUBEKPIMEMBER(connection, kpi_name, kpi_property, optional caption)

Connection is the text string of the name of the connection to the cube.

Kpi_name is the text string of the KPI in the cube.

Kpi_property is the component to be returned, such as the actual value (**KPIValue**), the target value (**KPIGoal**), state of the KPI (**KPIStatus**), value over time (**KPITrend**), or importance of the KPI (**KPIWeight**).

Caption is an optional argument for any alternative text string that is displayed in Excel instead of the **Kpi_name** and **Kpi_property**.

Errors to avoid

Remember that **CUBEKPIMEMBER** will only work with a SQL database that is connected to the worksheet. The cell will display **#GETTING_DATA** up until Excel pulls the requested information from the SQL database.

CUBEMEMBERPROPERTY function

The **CUBEMEMBERPROPERTY** pulls a particular "property" designated by a text string in Excel. Again, like the other **CUBE** functions, it is intended to create a flexible alternative to pivot tables.

Uses and applications

CUBEMEMBERPROPERTY pulls a particular property from a data set in Excel—the property is just a variable that is identified by a particular text string. For example, we could pull the size of a particular office location from a larger dataset of characteristics related to a set of office locations.

The formula

=CUBEMEMBERPROPERTY(connection, member-expression, Property)

Connection is the text string of the name of the connection to the cube.

Member_expression is the text string of a **multidimensional expression** (**MDX**) that refers to a unique member in the cube or a cell range.

Property is a text string of the name of the property that should be returned by the function from the data set. Essentially, it is the name of what you want the cell to return, such as "Office Size."

Errors to avoid

Again, note that **CUBEMEMBERPROPERTY** will only work if you have set up a data table for the function to pull from.

CUBERANKEDMEMBER function

The **CUBERANKEDMEMBER** function pulls a particular value based on rankings, like the salesperson with the third highest sales in a dataset or the top-ranked store location for a company.

Uses and applications

CUBERANKEDMEMBER can be used to identify which value meets certain rank criteria for data such as sales performance, grades in a classroom, the size of various locations, and more. It is primarily useful when dealing with large sets of data and gives the user the ability to do things such as pull the top five performers by some metric in that data.

The formula

=CUBERANKEDMEMBER(connection, set_expression, rank, optional caption)

Connection is the text string of the name of the connection to the cube.

Set_expression is a text string of a set expression identifying the values of interest.

`Rank` is the integer value of the rank in the **`set_expression`** to be returned. For instance, 3 would return the third highest observation for the given **`set_expression`**.

`Caption` is an optional argument for any alternative text string that is displayed in Excel.

Errors to avoid

If the connection name is not a valid workbook connection stored in the workbook, **`CUBERANKEDMEMBER`** returns a #NAME? error value.

CUBEVALUE function

The **`CUBEVALUE`** function pulls a summed value from another data set called a cube. Essentially, it is similar to using a pivot table, although it allows more flexibility in the Excel model setup.

Uses and applications

`CUBEVALUE` is useful for pulling the value of a particular element in a data cube. Essentially, if you are looking to add up sales or profits for some subset of a data set—similar to what might be done with a **`SUMIF`** function but in a more elegant framework—then the **`CUBEVALUE`** function is the best option.

The formula

=CUBEVALUE(connection, optional member expression)

`Connection` is the text string of the name of the connection to the cube.

`Member_expression` is a text string of a member expression, identifying the variable of interest that you want to sum up. While it is optional, you generally want to include it because it lets you subset the data to find a particular set of values you want to sum up—if it is not included, the values for the whole cube would be summed.

Errors to avoid

`CUBEVALUE` has all of the complexities of the other **`CUBE`** function. As long as you define your cube correctly and pick the relevant **`member_expression`**, you will be fine.

Conclusion

The functions shown in this chapter are complex specialized tools that serve as a more flexible and elegant alternative to the better-known pivot tables. These are truly advanced functions—if you want to build an impressive Excel model that will impress your colleagues and provide a useful output based on a dataset, the cube functions are great ones to master.

Multiple choice questions

1. **Cube functions are an alternative to what other popular Excel tools?**
    a. Excel Add-ins
    b. Pivot tables
    c. Summary plots
    d. None of the above

2. **Cube functions all rely on a link to a separate set of data called what?**
    a. Set
    b. Connection
    c. Link_time
    d. Rank

3. **If you wanted to pull a series of the top five performers from a dataset of salespeople, which function would you most likely use?**
    a. =CUBESET
    b. =CUBEKPI
    c. =CUBEVALUE
    d. =CUBERANKEDMEMBER

Answers

1. **b**
2. **b**
3. **d**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 13

Web Functions in Excel

Introduction

Excel does not have many of the more sophisticated tools of other software packages that interact and pull information from the Web, but it does have some specific tools that you should be familiar with. In this chapter, we will cover the most popular and useful formulas that you need to know to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- ENCODEURL function
- FILTERXML function
- WEBSERVICE function

Objectives

After studying this unit, you will be able to use the limited set of Web functions available in Excel.

ENCODEURL function

The **ENCODEURL** function is a fast tool to turn any expression in Excel into a URL-encoded string, where non-alphanumeric characters are replaced with the percentage (%) symbol and a hexadecimal number.

Uses and applications

ENCODEURL is useful for either pulling information from Excel files shared on a cloud-based server (for example, Microsoft Sharepoint) or for use in combination with the **WEBSERVICE** function to pull information such as stock ticker data off of the internet.

The formula

=ENCODEURL(reference string to be encoded)

As *figure 13.1* illustrates, **ENCODEURL** will encode any string of characters as a URL. This is not useful by itself, but it is useful when used with a Web API, for pulling data, or for pulling values from another Excel sheet in Sharepoint.

	A	B	C
1	Input	Output	Formula
2	Excel	Excel	=ENCODEURL(A2)
3	www.Google.com	www.Google.com	=ENCODEURL(A3)
4	D1:D30	D1%3AD30	=ENCODEURL(A4)
5	SUM(D1:D30)	SUM%28D1%3AD30%29	=ENCODEURL(A5)

Figure 13.1: *The ENCODEURL function in excel*

Errors to avoid

ENCODEURL is simple to use, but remember that it is really just an intermediate function that relies on other functions to be useful in a model. The function is only available in Excel 2013 and later and does not work with Web-based Excel.

FILTERXML function

The **FILTERXML** function allows users to pull data from XML text based on an **xpath** expression.

Uses and applications

XML is often used for structuring information in databases. The **FILTERXML** function enables the Excel user to pull data from XML and represent it in Excel. This can be very useful, especially when combined with the **WEBSERVICE** and **ENCODEURL** functions.

The formula

=FILTERXML(Xml, xpath)

As *figure 13.2* shows, if given XML text, it is easy to turn that data into plain output using the **FILTERXML** function. In the middle column, the **FILTERXML** function has been used to pull the "title" example from the XMLtags **<title>** and **</title>**.

XML is the location of the XML data. In this case, it is cell A7, but it could also be a remote link, and **xpath** is the path to the information to be pulled. Here, it is under the title label. Refer to *figure 13.2:*

6			
7	<title> Example </title>	Example	=FILTERXML(A7, "/title")

Figure 13.2: *The FILTERXML function in Excel*

Errors to avoid

FILTERXML is fairly straightforward to use, but be aware that it is only available for Excel on Windows and will not work with Excel on Mac.

WEBSERVICE function

The **WEBSERVICE** function enables you to pull information off of the internet and into Excel.

Uses and applications

WEBSERVICE typically pulls information in XML format, which is why it pairs well with the **FILTERXML** function. For instance, stock price data or breaking news information can be pulled into Excel using the **WEBSERVICE** function.

The formula

=WEBSERVICE(URL web link to pull information from)

Errors to avoid

The **WEBSERVICE** function is a little bit tricky—it takes some trial and error to figure out how to display the information you are looking for, so be sure to make adjustments depending on which Web service you are calling the information from.

Conclusion

The functions shown in this chapter are complex, specialized tools that will allow you to pull in information from the internet directly into Excel. These are powerful functions that can let you build complex models with useful outcomes if you master them. Take the time to explore each function fully, and you will find significant uses for them in almost any business area.

Multiple choice questions

1. **The ENCODE function replaces non-alphanumeric characters with what?**
    a. The % sign
    b. The ! sign
    c. The letter A
    d. None of the above

2. **Which function can be used to pull information from the internet into Excel?**
    a. WEBSERVICE
    b. WEBLINK
    c. XMLFILTER
    d. DATAGATHER

3. **What function enables pulling data from XML datasets?**
    a. XML.TEXT
    b. XMLFILTER
    c. CUBEVALUE
    d. CUBEXML

Answers

1. **a**
2. **a**
3. **b**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Chapter 14

User-defined Functions in Excel

Introduction

User-defined functions are advanced but very powerful functions that can enable Excel to greatly speed up tedious and time-consuming tasks. Although these functions are harder to use, they can be a huge time-saver for people doing specific kinds of projects. In this chapter, we will cover the most popular and useful formulas that you need to know in order to be able to use Excel effectively.

Structure

In this chapter, we will discuss the following topics:

- CALL function
- EUROCONVERT function
- REGISTER.ID function

Objectives

After studying this unit, you will be able to start creating user-defined functions in Excel. These are all advanced functions that should be used with caution as they can cause damage to your computer's operating system if used incorrectly.

CALL function

The **CALL** function is a function for macro-enabled worksheets, which calls a procedure from a dynamic link library or code resource. Essentially, you can use this function to pull in information from outside resources.

NOTE: Incorrectly editing the registry may severely damage your operating system, requiring you to reinstall it. Microsoft cannot guarantee that problems resulting from editing the registry incorrectly can be resolved. Before editing the registry, back up any valuable data. For the most recent information about using and protecting your computer's registry, see Microsoft Windows Help.

Uses and applications

The **CALL** function is very useful, although practically speaking, it is not an easy function to set up. You must be using a macro-enabled Excel workbook (which is generally more unstable) and then link that workbook to an outside **Dynamic Link Library** (**DLL**) that contains the procedure in Microsoft Excel for Windows. This process requires you to have the function already created in the DLL. Hence, it is most useful when dealing with a repetitive task that must be done routinely where you pull a procedure from an outside resource.

The formula

The formula for the **CALL** function is as follows:

CALL(module_text, procedure, type_text, optional arguments)

The CALL function syntax has the following arguments:

- **Module_text**: This is required text in quotes, and it specifies the name of the dynamic link library that contains the procedure in question.
- **Procedure**: This is text specifying the name of the function in the DLL.
- **Type_text**: This is text specifying the data type of the return value and the data types of all arguments to the DLL or code resource. The first letter of **type_text** specifies the return value.
- Optional arguments: These arguments are to be passed to the procedure.

Errors to avoid

Keep in mind that the **CALL** function is potentially risky to use, especially when used in combination with the **REGISTER** function. Beyond that, the biggest error made

typically comes when the user tries to call a procedure that does not exist. In this case, the function will not be able to execute the procedure.

EUROCONVERT function

The **EUROCONVERT** function is a special function that relies on an Excel Add-in. It is emblematic of the many additional Excel functions that become available once you start using add-ins in Excel.

Uses and applications

To use the **EUROCONVERT** add-in, you need to have the Euro Currency tools add-in installed in Excel. Excel Add-ins are powerful tools that can build new functionality into Excel. Excel starts with several of these Add-ins on its own, but hundreds more can be downloaded and added through the Excel store (under the Insert tab).

To insert add-ins such as Euro Currency tools, go to the **File** | **Options** | **Add-ins**. Then click on the Go button next to "Manage: Excel Add-ins" at the bottom. This will bring you to a menu of available Excel Ad-ins that have been downloaded and are ready to be installed in Excel, as *figure 14.1* illustrates:

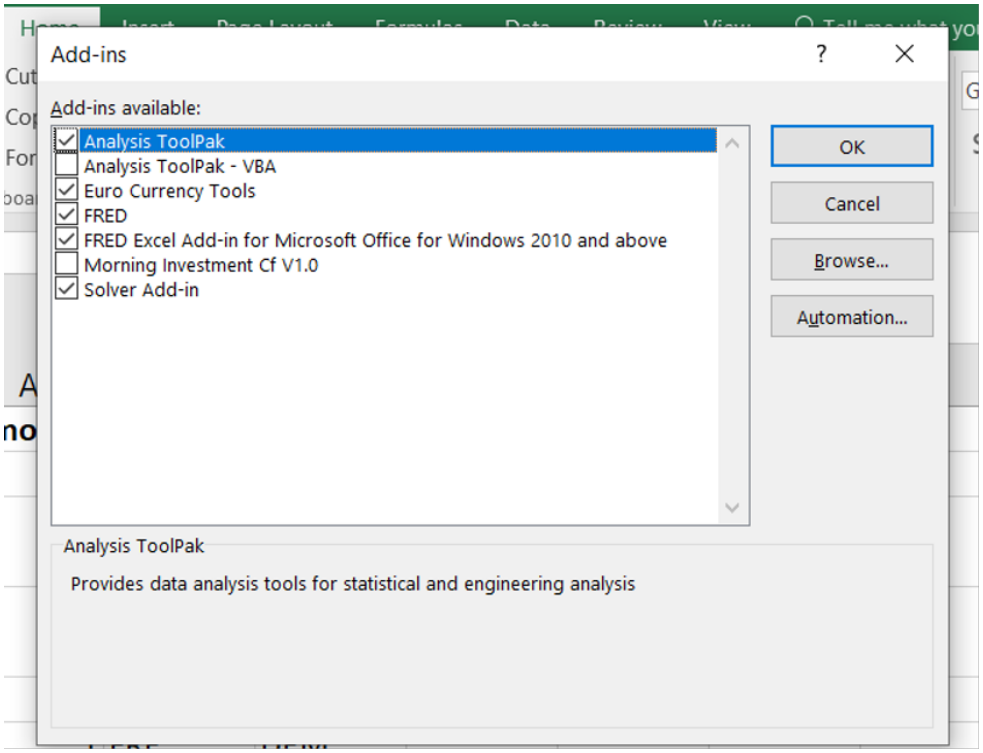


Figure 14.1: *Inserting Excel add-ins*

The formula

The formula for the **EUROCONVERT** function is as follows:

EUROCONVERT(Amount of currency to be converted to another currency, beginning currency, ending/conversion currency)

The **EUROCONVERT** function syntax has the following arguments:

- The amount of currency to be converted is simply a number indicating the amount of Pounds, Euros, Francs, or other currency to be converted.
- The beginning currency is the original currency type that the amount starts in.
- The ending currency is the type of currency for the starting currency to be converted into.

Figure 14.2 gives a list of the different types of currencies that can be used with **EUROCONVERT**. With the change in the European economy (and associated currencies) over the past two decades, this list is somewhat archaic, but the function illustrates the usefulness of Excel Add-ins more broadly.

Country/Region	Basic unit of currency	ISO code
Belgium	franc	BEF
Luxembourg	franc	LUF
Germany	deutsche mark	DEM
Spain	peseta	ESP
France	franc	FRF
Ireland	pound	IEP
Italy	lira	ITL
Netherlands	guilder	NLG
Austria	schilling	ATS
Portugal	escudo	PTE
Finland	markka	FIM
Greece	drachma	GRD
Slovenia	tolar	SIT
Euro member states	euro	EUR

Figure 14.2: *The EUROCONVERT function works with these currencies*

Figure 14.3 illustrates the use of the **EUROCONVERT** add-in. Here, **DEM** is being converted to Euros. Notice that this works even when a "$" is added in front of the number, as in the lower example, which relies on D5.

	A	B	C	D	E
1	Amount	Source	Target		
2	1.2	DEM	EUR		
3	1	FRF	EUR		
4	1	FRF	EUR		
5	1	FRF	DEM	$5	
6	1	FRF	DEM		
7					
8					
9		0.61			
10		=EUROCONVERT(A2,B2,C2)			
11		2.56			
12		=EUROCONVERT(D5, "DEM", "EUR")			

Figure 14.3: *The EUROCONVERT function is an example of the type of additional functions that can be added with Excel add-ins*

Errors to avoid

Broadly speaking, **`EUROCONVERT`** is easy to use, but the bigger point here is the usefulness of Excel Add-ins. These Add-ins can be developed by third parties and offered through the Excel store, or they can even be developed for personal use by the Excel user themselves. With such Add-ins, Excel gains considerable power and flexibility for a variety of specialized scenarios.

REGISTER.ID function

The **`REGISTER.ID`** function returns the register ID of the specified DLL or code resource if it has been previously registered (or registers the DLL or code resource and then returns the register ID if it has not been previously registered). **`REGISTER.ID`** can be used on worksheets (unlike **`REGISTER`**), but you cannot specify a function name and argument names with **`REGISTER.ID`**.

NOTE: Incorrectly editing the registry may severely damage your operating system, requiring you to reinstall it. Microsoft cannot guarantee that problems resulting from editing the registry incorrectly can be resolved. Before editing the registry, back up any valuable data. For the most recent information about using and protecting your computer's registry, see Microsoft Windows Help.

Uses and applications

REGISTER.ID (and **REGISTER**) are primarily used with the **CALL** function to pull in a specified procedure from an outside resource. As noted with the **CALL** function, this process requires you to have the function already created in the **DLL**. Hence, it is most useful when dealing with a repetitive task that must be done routinely, where you pull a procedure from an outside resource.

The formula

The formula for the **REGISTER.ID** function is as follows:

REGISTER.ID(module_text, procedure, type_text)

The **REGISTER.ID** function syntax has the following arguments:

- **Module_text** is required text in quotes, which specifies the name of the dynamic link library that contains the function.
- **Procedure** is text specifying the name of the function in the DLL.
- Optional **Type_text** is text specifying the data type of the return value and the data types of all arguments to the DLL or code resource. The first letter of **type_text** specifies the return value. If the function or code is already registered, this can be omitted.

Errors to avoid

As before, keep in mind that the **REGISTER** function is risky to use. Be very careful about registering DLLs and using these functions in general. Make sure you understand what you are doing before executing these functions to avoid significant trouble with having to reinstall your operating system.

Conclusion

The functions shown in this chapter are very advanced Excel tools that can create powerful opportunities to accomplish specialized tasks in Excel. Add-ins and the functions associated with them are something that can benefit almost all Excel users and carry minimal cost and risk, whereas the **REGISTER** function and associated **CALL** function are something you should be more careful with.

This chapter takes us to the end of our discussion of Excel functions. Regardless of your specific use cases for Excel, it is likely you will want to go further in your learning and begin practicing building Excel models based on these functions. There are a number of great resources to do this, such as the new LinkedIn Learning

courses available through LinkedIn. Beyond that, there is really no substitute for the experience of building a model on your own, and so the best way to become proficient in Excel is often to learn by doing. Start with the task you need to accomplish and ask yourself what tools in Excel and functions from this book can help you accomplish that task. You will quickly find that with a little practice, you will begin building effective Excel models that will impress your coworkers and colleagues, and others will begin turning to you for advice and assistance in using Excel for their own computational models.

I wish you the best of luck and professional success going forward, and I hope you have found this text to be useful in your professional journey.

Multiple choice questions

1. **EUROCONVERT is an example of an Excel function that works with what additional supplement?**
    a. An Excel Add-in
    b. Excel VBA Libraries
    c. Currency formulas
    d. None of the above

2. **The CALL function works with what outside resource?**
    a. Dynamic link library or DLL
    b. Active Web links or AWL
    c. Extensible markup language or XML
    d. The internet

3. **True or False, Using the CALL function and REGISTRY function can be harmful to your computer if you do not understand how to use them and incorrectly edit your computer's registry?**
    a. FALSE
    b. TRUE

Answers

1. **a**
2. **a**
3. **b**

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Index

A
ABS function
about 55
errors, avoiding 55
formula 55
usage and applications 55
ACOS function
about 55
errors, avoiding 56
formula 56
usage and applications 55
ACOT function
about 56
errors, avoiding 57
formula 57
usage and applications 56
ACOTH function
about 55, 56
errors, avoiding 56, 57
formula 56, 57
usage and applications 55, 56
ADDRESS function
about 34
errors, avoiding 35
formula 34, 35
usage and applications 34
AND function
about 22
errors, avoiding 23
formula 22
usage and application 22
AREAS function
about 35
errors, avoiding 36
formula 35, 36
usage and applications 35

ASIN function
about 57
errors, avoiding 58
formula 58
usage and applications 57
ASINH function
about 57
errors, avoiding 58
formula 58
usage and applications 57
ATAN function
about 58
errors, avoiding 59
formula 59
usage and applications 58
ATANH function
about 58
errors, avoiding 59
formula 59
usage and applications 58

C

CALL function
about 208
errors, avoiding 208
formula 208
usage and applications 208
CEILING function
about 59
errors, avoiding 60
formula 59
usage and applications 59
Cell function
about 136
errors, avoiding 137
formula 136, 137
usage and applications 136
CHOOSE function
about 14
errors, avoiding 14
formula 14
usage and applications 14
CLEAN function
about 116
errors, avoiding 117
formula 117
usage and applications 116
CODE function
about 117
errors, avoiding 118
formula 117
usage and applications 117
COLUMN function
about 36
errors, avoiding 37
formula 36
usage and application 36
COLUMNS function
about 36
errors, avoiding 37
formula 36
usage and applications 36
COMBINA function
about 60
errors, avoiding 61
formula 60, 61
usage and applications 60
COMBIN function
about 60
errors, avoiding 61
formula 60, 61
usage and applications 60

CONCATENATE function
about 118
errors, avoiding 118
formula 118
usage and applications 118
COS function
about 61
errors, avoiding 62
formula 61
usage and applications 61
COSH function
about 61
errors, avoiding 62
formula 61
usage and applications 61
COT function
about 62
errors, avoiding 63
formula 62
usage and applications 62
COTH function
about 62
errors, avoiding 63
formula 62
usage and applications 62
COUPDAYS function
about 152
errors, avoiding 153
formula 153
usage and applications 152
COUPNUM function
about 152
errors, avoiding 153
formula 153
usage and applications 152
COUPPCD function
about 152
errors, avoiding 153
formula 153
usage and applications 152
CSC function
about 63
errors, avoiding 64
formula 63
usage and applications 63
CSCH function
about 63
errors, avoiding 64
formula 63
usage and applications 63
CUBEKPIMEMBER function
about 197
errors, avoiding 197
formula 197
usage and applications 197
CUBEMEMBER function
about 196
errors, avoiding 197
formula 196
usage and applications 196
CUBEMEMBERPROPERTY function
about 197
errors, avoiding 198
formula 198
usage and applications 198
CUBERANKEDMEMBER function
about 198
errors, avoiding 199
formula 198
usage and applications 198

CUBEVALUE function
about 199
errors, avoiding 199
formula 199
usage and applications 199
CUMIPMT function
about 153
errors, avoiding 154
formula 154
usage and applications 154
CUMPRINC function
about 153
errors, avoiding 154
formula 154
usage and applications 154
Cumulative Density Function (CDF) 87

D

DATE function
about 14, 172
errors, avoiding 15, 173
formula 15, 172
usage and applications 15, 172
DAVERAGE function
about 184
errors, avoiding 185
formula 184
usage and applications 184
DAY, MONTH, and YEAR function
about 173
errors, avoiding 173
formula 173
usage and applications 173
DAYS and DAYS360 function
about 174
errors, avoiding 174
formula 174
usage and applications 174
DAYS function
about 15
errors, avoiding 16
formula 16
usage and applications 15
DB function
about 155
formula 155
usage and applications 155
DCOUNTA function
about 186
errors, avoiding 187
formula 186
usage and applications 186
DCOUNT function
about 185
errors, avoiding 186
formula 185
usage and applications 185
DDB function
about 155
errors, avoiding 155
formula 155
usage and applications 155
DECIMAL function
about 64
errors, avoiding 65
formula 64
usage and applications 64
declining balance (DB) method 155
DEGREES function
about 65
errors, avoiding 66
formula 65
usage and applications 65

DGET function
about 187
errors, avoiding 188
formula 187
usage and applications 187
DISC function
about 156
errors, avoiding 157
formula 156
usage and applications 156
DMIN and DMAX function
about 188
errors, avoiding 189
formula 188
usage and applications 188
DOLLAR function
about 118
errors, avoiding 119
formula 119
usage and application 119
double declining balance (DDB) method 155
DPRODUCT function
about 189
errors, avoiding 190
formula 189, 190
usage and applications 189
DSTDEV and DVAR function
about 191
errors, avoiding 192
formula 192
usage and applications 191
DSUM function
about 190
errors, avoiding 191
formula 190, 191
usage and applications 190
DURATION function
about 157
errors, avoiding 158
formula 157
usage and applications 157
Dynamic Link Library (DLL) 45

E

EDATE and EOMONTH function
about 174
errors, avoiding 175
formula 175
usage and applications 175
EFFECT function
about 158
errors, avoiding 158
formula 158
usage and applications 158
ENCODEURL function
about 202
errors, avoiding 202
formula 202
usage and applications 202
ERROR.TYPE function
about 137
errors, avoiding 138
formula 137, 138
usage and applications 137
EUROCONVERT function
about 209
arguments 210
errors, avoiding 211
formula 210
usage and applications 209

EVEN function
about 66
errors, avoiding 66
formula 66
usage and applications 66
EXACT function
about 119
errors, avoiding 120
formula 120
usage and applications 119
Excel
using 2-4

F

FALSE function
about 23
errors, avoiding 24
formula 24
usage and applications 23
FILTER function
about 37
errors, avoiding 37
formula 37
usage and applications 37
FILTERXML function
about 203
errors, avoiding 203
formula 203
usage and applications 203
FINDB functions
about 16
errors, avoiding 16, 17
formula 16
usage and applications 16
FIND function
about 16, 120
errors, avoiding 121
formula 120
usage and applications 16, 120
FIXED function
about 121
errors, avoiding 121
formula 121
usage and applications 121
FLOOR function
about 59
errors, avoiding 60
formula 59
usage and applications 59
FORECAST function
about 97
errors, avoiding 98
formula 97
usage and applications 97
FORECAST.LINEAR function
about 97
errors, avoiding 98
formula 97
usage and applications 97
FORMULATEXT function
about 38
errors, avoiding 38
formula 38
usage and applications 38

G

GEOMEAN function
about 98
errors, avoiding 99
formula 99
usage and applications 98
GETPIVOTDATA function
about 38
errors, avoiding 39

formula 39
usage and applications 38
GROWTH function
about 99
errors, avoiding 100
formula 99
usage and applications 99

H

HARMEAN function
about 98
errors, avoiding 99
formula 99
usage and applications 98
HLOOKUP function
about 40
errors, avoiding 40
formula 40
usage and applications 40
HOUR, MINUTE, and SECOND function
about 175
errors, avoiding 176
formula 176
usage and applications 175
HYPERLINK function
about 41
errors, avoiding 41
formula 41
usage and applications 41

I

IFERROR function
about 25
errors, avoiding 26
formula 25
usage and applications 25
IF function
about 8
errors, avoiding 9, 10
formula 9
usage and application 8
IFNA function
about 26
errors, avoiding 27
formula 26
usage and application 26
IFS function
about 27
errors, avoiding 28
formula 27
usage and applications 27
INDEX function
about 17
errors, avoiding 17, 18
formula 17
usage and applications 17
INDIRECT function
about 41
errors, avoiding 43
formula 42
usage and applications 42
INFO function
about 138
errors, avoiding 139
formula 138
usage and applications 138
INT function
about 67
errors, avoiding 67
formula 67
usage and applications 67

INTRATE function
about 160
errors, avoiding 161
formula 160
usage and applications 160
IPMT function
about 161
errors, avoiding 162
formula 161
usage and applications 161
IRR function
about 162
errors, avoiding 163
formula 162, 163
usage and applications 162
ISBLANK function
about 139
errors, avoiding 139
formula 139
usage and applications 139
ISERR function
about 140
errors, avoiding 140
formula 140
usage and applications 140
ISERROR function
about 140
errors, avoiding 140
formula 140
usage and applications 140
ISEVEN function
about 140
errors, avoiding 141
formula 141
usage and applications 141
ISFORMULA function
about 141
errors, avoiding 142
formula 142
usage and applications 141
ISLOGICAL function
about 142
errors, avoiding 142
formula 142
usage and applications 142
ISNA function
about 143
errors, avoiding 143
formula 143
usage and applications 143
ISNUMBER function
about 143
errors, avoiding 144
formula 144
usage and applications 143
ISODD function
about 140
errors, avoiding 141
formula 141
usage and applications 141
ISOWEEKNUM and WEEKNUM function
about 176
errors, avoiding 177
formula 177
usage and applications 177
ISREF function
about 144
errors, avoiding 145
formula 144
usage and applications 144

ISTEXT function
about 145
errors, avoiding 145
formula 145
usage and applications 145

L

LCM function
about 67
errors, avoiding 68
formula 68
usage and applications 68
LEFT and RIGHT function
about 121
errors, avoiding 122
formula 122
usage and applications 122
LEN function
about 122
errors, avoiding 123
formula 123
usage and applications 123
LN function
about 68
errors, avoiding 69
usage and applications 68
LOG function
about 68
errors, avoiding 69
formula 68
usage and applications 68
LOGNORM.DIST function
about 100
errors, avoiding 101
formula 100, 101
usage and applications 100
LOGNORM.INV function
about 100
errors, avoiding 101
formula 100, 101
usage and applications 100
LOOKUP function
about 10
errors, avoiding 10, 11
formula 10
usage and application 10
LOWER function
about 123
errors, avoiding 124
formula 124
usage and applications 123

M

MATCH function
about 13
errors, avoiding 13
formula 13
usage and applications 13
MDURATION function
about 157
errors, avoiding 158
formula 157
usage and applications 157
MEDIAN function
about 101
errors, avoiding 102
formula 102
usage and applications 101
MID function
about 124
errors, avoiding 125
formula 124
usage and applications 124

MIN function
about 102
errors, avoiding 103
formula 103
usage and applications 102
MIRR function
about 162, 163
errors, avoiding 163
usage and applications 162
MMULT function
about 69
errors, avoiding 70
formula 69
usage and applications 69
MOD function
about 70
errors, avoiding 70
formula 70
usage and applications 70

N

NEGBINOM.DIST function
about 103
errors, avoiding 104
formula 103
usage and applications 103
NETWORKDAYS function
about 177
errors, avoiding 178
formula 178
usage and applications 177, 178
N function
about 145
errors, avoiding 146
formula 146
usage and applications 146
NOMINAL function
about 163
errors, avoiding 164
formula 164
usage and applications 163, 164
NORM.DIST function
about 86
errors, avoiding 87
formula 87
usage and applications 87
NORM.INV function
about 86
errors, avoiding 87
formula 87
usage and applications 87
NOT function
about 28
errors, avoiding 29
formula 28
usage and application 28
NOW and TODAY function
about 178
errors, avoiding 179
formula 179
usage and applications 178
NPV function
about 162
errors, avoiding 163
formula 162, 163
usage and applications 162
number of periods required in a financial problem (NPER) 159
NUMBERVALUE function
about 125
errors, avoiding 125
formula 125

usage and applications 125

O

ODD function
about 66
errors, avoiding 66
formula 66
usage and applications 66
OFFSET function
about 43
errors, avoiding 44
formula 44
usage and applications 44
OR function
about 29
errors, avoiding 29
formula 29
usage and applications 29

P

PERCENTILE.EXC function
about 88
formula 88
usage and applications 88
PERCENTILE.INC function
about 88
errors, avoiding 89
formula 88
usage and applications 88
PHI function
about 89
errors, avoiding 89
formula 89
usage and applications 89
POISSON.DIST function
about 90
errors, avoiding 90
formula 90
usage and applications 90
POWER function
about 71
errors, avoiding 71
formula 71
usage and applications 71
PRICE function
about 164
errors, avoiding 165
formula 165
usage and applications 164
PROB function
about 90
errors, avoiding 91
formula 91
usage and applications 91
PRODUCT function
about 71
errors, avoiding 72
formula 72
usage and applications 72
PROPER function
about 125
errors, avoiding 126
formula 126
usage and applications 126

Q

QUARTILE.EXC function
about 92
errors, avoiding 93
formula 92
usage and applications 92
QUARTILE.INC function
about 92
errors, avoiding 93

formula 92
usage and applications 92

R

RADIANS function
about 65
errors, avoiding 66
formula 65
usage and applications 65
RANDBETWEEN function
about 72
errors, avoiding 73
formula 73
usage and applications 72
RAND function
about 72
errors, avoiding 73
formula 73
usage and applications 72
RANK.AVG function
about 93
errors, avoiding 94
formula 93
usage and applications 93
RANK.EQ function
about 93
errors, avoiding 94
formula 93
usage and applications 93
REGISTER.ID function
about 211
arguments 212
errors, avoiding 212
formula 212
usage and applications 212
REPLACE function
about 126
errors, avoiding 127
formula 126, 127
usage and applications 126
ROMAN function
about 73
errors, avoiding 74
formula 74
usage and applications 74
ROUNDDOWN function
about 75
errors, avoiding 76
formula 75
usage and applications 75
ROUND function
about 75
errors, avoiding 76
formula 75
usage and applications 75
ROUNDUP function
about 75
errors, avoiding 76
formula 75
usage and applications 75
ROW function
about 44
errors, avoiding 45
formula 44, 45
usage and applications 44
ROWS function
about 44
errors, avoiding 45
formula 44, 45
usage and applications 44
RSQ function
about 94
errors, avoiding 94

formula 94
usage and applications 94
R-squared 94
RTD function
about 45
errors, avoiding 46
formula 46
usage and applications 45

S

SEARCH function
about 127
errors, avoiding 128
formula 127
usage and applications 127
SEC function
about 76
errors, avoiding 76
formula 76
usage and applications 76
SECH function
about 76
errors, avoiding 76
formula 76
usage and applications 76
SHEET function
about 146
errors, avoiding 147
formula 147
usage and applications 146
SIGN function
about 77
errors, avoiding 77
formula 77
usage and applications 77
SIN function
about 77
errors, avoiding 78
formula 78
usage and applications 77
SINH function
about 77
errors, avoiding 78
formula 78
usage and applications 77
SKEW function
about 95
errors, avoiding 96
formula 95
usage and applications 95
SKEW.S function
about 95
errors, avoiding 96
formula 95
usage and applications 95
SORTBY function
about 46
errors, avoiding 47
formula 47
usage and applications 46
SORT function
about 46
formula 46, 47
usage and applications 46
SQRT function
about 78
errors, avoiding 79
formula 78, 79
usage and applications 78
STDEVA function
about 96
errors, avoiding 97
formula 96

usage and applications 96
STDEVPA function
about 96
errors, avoiding 97
formula 96
usage and applications 96
STEYX function
about 104
errors, avoiding 105
formula 104
usage and applications 104
SUBSTITUTE function
about 128
errors, avoiding 128
formula 128
usage and applications 128
SUBTOTAL function
about 79
errors, avoiding 80
formula 79, 80
usage and applications 79
SUM function
about 6
errors, avoiding 7, 8
formula 6
usage and applications 6
SUMIF function
about 80
errors, avoiding 81
formula 81
usage and applications 80
SUMPRODUCT function
about 80
errors, avoiding 81
formula 81
usage and applications 80
SUMSQ function
about 80
errors, avoiding 81
formula 81
usage and applications 80
SWITCH function
about 30
errors, avoiding 31
formula 30
usage and applications 30

T

TAN function
about 81
formula 82
usage and applications 81
TANH function
about 81
errors, avoiding 82
formula 82
usage and applications 81
T.DIST.2T function
about 105
errors, avoiding 106
formula 105
usage and applications 105
T.DIST function
about 105
errors, avoiding 106
formula 105
usage and applications 105
T.DIST.RT function
about 105
errors, avoiding 106
formula 105
usage and applications 105

TEXT function
about 129
errors, avoiding 130
formula 129
usage and applications 129
T function
about 128
errors, avoiding 129
formula 129
usage and applications 129
TIME function
about 179
errors, avoiding 179
formula 179
usage and applications 179
time value of money function
about 158
errors, avoiding 160
formula 159
usage and applications 158, 159
T.INV.2T function
about 106
errors, avoiding 107
formula 106
usage and applications 106
T.INV function
about 106
errors, avoiding 107
formula 106
usage and applications 106
TRANSPOSE function
about 47
errors, avoiding 48
formula 48
usage and applications 47
TREND function
about 108
errors, avoiding 109
formula 108
usage and applications 108
TRIM function
about 130
errors, avoiding 130
formula 130
usage and applications 130
TRIMMEAN function
about 109
errors, avoiding 110
formula 109
usage and applications 109
TRUE function
about 24
errors, avoiding 25
formula 24
usage and applications 24
T.TEST function
about 107
errors, avoiding 108
formula 107
usage and applications 107
Type function
about 147
errors, avoiding 148
formula 148
usage and applications 147

U

UNICHAR function
about 130
errors, avoiding 131

formula 131
usage and applications 131
UNICODE function
about 131
errors, avoiding 132
formula 131
usage and applications 131
UNIQUE function
about 48
errors, avoiding 49
formula 49
usage and applications 48
UPPER function
about 123
errors, avoiding 124
formula 124
usage and applications 123

V

VALUE function
about 132
errors, avoiding 132
formula 132
usage and applications 132
VAR.P function
about 110
formula 110, 111
usage and applications 110
VAR.S function
about 110
errors, avoiding 111
formula 110, 111
usage and applications 110
VDB function
about 167
errors, avoiding 168
formula 167
usage and applications 167
VLOOKUP function
about 11
errors, avoiding 12
formula 12
usage and application 11

W

WEBSERVICE function
about 204
errors, avoiding 204
formula 204
usage and applications 204
WEEKDAY function
about 180
errors, avoiding 180
formula 180
usage and applications 180

X

XIRR function
about 166
errors, avoiding 167
formula 166
usage and applications 166
XLOOKUP function
about 49
errors, avoiding 50
formula 49, 50
usage and applications 49
XMATCH function
about 50
errors, avoiding 50
formula 50
usage and applications 50
XNPV function
about 166

errors, avoiding 167
formula 166
usage and applications 166
XOR function
about 30
errors, avoiding 30
formula 30
usage and application 30

Y

YIELDDISC function
about 164
errors, avoiding 165
formula 165
usage and applications 164
YIELD function
about 164
errors, avoiding 165
formula 165
usage and applications 164
YIELDMAT function
about 164
errors, avoiding 165
formula 165
usage and applications 164

Z

Z.TEST function
about 111
errors, avoiding 112
formula 111
usage and applications 111