

Financial Modeling

Simon Benninga

fourth edition



A blurred image of a financial table with numbers and percentages. The table has two columns. The left column contains numbers: 129, 266, 1429, 88457, 71729, 188457, 171429, 188457, 188457, 188457, 188457. The right column contains percentages: 0.7, 2.9, 2.9, 0.7, 2.9. The numbers in the left column are repeated, and the percentages in the right column are also repeated.

129	0.7
266	2.9
1429	2.9
88457	
71729	0.7
188457	
171429	2
188457	
188457	
188457	
188457	
188457	

FINANCIAL MODELING

FINANCIAL MODELING

Simon Benninga

With a section on Visual Basic for Applications
by Benjamin Czaczk

FOURTH EDITION

The MIT Press
Cambridge, Massachusetts
London, England

© 2014 Massachusetts Institute of Technology

All rights reserved. No part of this book may be reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

MIT Press books may be purchased at special quantity discounts for business or sales promotional use. For information, please email special_sales@mitpress.mit.edu.

Print copies of this book include a card affixed to the inside back cover with a unique access code. Access codes are required to download Excel worksheets and solutions to end-of-chapter exercises. If you purchased an e-book, you may obtain a unique access code by emailing journals-access@mit.edu or calling 617-253-2889 or 800-207-8354 (toll-free in the U.S. and Canada).

This book was set in Times Roman by Toppan Best-set Premedia Limited. Printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data

Benninga, Simon.

Financial modeling / Simon Benninga.—Fourth edition.

pages cm

Includes bibliographical references and index.

ISBN 978-0-262-02728-1 (hardcover : alk. paper)

1. Finance—Mathematical models. 2. Microsoft Visual Basic for applications. I. Title.

HG173.B46 2014

332.01'5118—dc23

2013032409

10 9 8 7 6 5 4 3 2 1

To the memory of our parents:

Helen Benninga (1913–2008)

Groningen, Netherlands – Jerusalem, Israel

Noach Benninga (1909–1994)

Eenrum, Netherlands – Asheville, North Carolina

Esther Czaczkes (1931–2012)

Jerusalem, Israel – Jerusalem, Israel

Alfred Czaczkes (1923–1997)

Vienna, Austria – Jerusalem, Israel

Contents

Preface	xxi
Before All Else	1
0.1 Data Tables	1
0.2 What Is Getformula?	1
0.3 How to Put Getformula into Your Excel Notebook	1
0.4 Saving the Excel Workbook: Windows	4
0.5 Saving the Excel Workbook: Mac	5
0.6 Do You Have to Put Getformula into Each Excel Workbook?	6
0.7 A Shortcut to Use Getformula	6
0.8 Recording Getformula: The Windows Case	7
0.9 Recording Getformula: The Mac Case	10
I CORPORATE FINANCE AND VALUATION	11
1 Basic Financial Calculations	13
1.1 Overview	13
1.2 Present Value and Net Present Value	14
1.3 The Internal Rate of Return (IRR) and Loan Tables	20
1.4 Multiple Internal Rates of Return	27
1.5 Flat Payment Schedules	29
1.6 Future Values and Applications	30
1.7 A Pension Problem—Complicating the Future Value Problem	33
1.8 Continuous Compounding	38
1.9 Discounting Using Dated Cash Flows	42
Exercises	45
2 Corporate Valuation Overview	53
2.1 Overview	53
2.2 Four Methods to Compute Enterprise Value (EV)	53
2.3 Using Accounting Book Values to Value a Company: The Firm’s Accounting Enterprise Value	54
2.4 The Efficient Markets Approach to Corporate Valuation	58
2.5 Enterprise Value (EV) as the Present Value of the Free Cash Flows: DCF “Top Down” Valuation	60

2.6	Free Cash Flows Based on Consolidated Statement of Cash Flows (CSCF)	63
2.7	ABC Corp., Consolidated Statement of Cash Flows (CSCF)	64
2.8	Free Cash Flows Based on Pro Forma Financial Statements	67
2.9	Summary	69
	Exercises	70
3	Calculating the Weighted Average Cost of Capital (WACC)	71
3.1	Overview	71
3.2	Computing the Value of the Firm's Equity, E	73
3.3	Computing the Value of the Firm's Debt, D	74
3.4	Computing the Firm's Tax Rate, T_C	75
3.5	Computing the Firm's Cost of Debt, r_D	76
3.6	Two Approaches to Computing the Firm's Cost of Equity, r_E	82
3.7	Implementing the Gordon Model for r_E	82
3.8	The CAPM: Computing the Beta, β	89
3.9	Using the Security Market Line (SML) to Calculate Merck's Cost of Equity, r_E	96
3.10	Three Approaches to Computing the Expected Return on the Market, $E(r_M)$	98
3.11	What's the Risk-Free Rate r_f in the CAPM?	102
3.12	Computing the WACC, Three Cases	102
3.13	Computing the WACC for Merck (MRK)	103
3.14	Computing the WACC for Whole Foods (WFM)	104
3.15	Computing the WACC for Caterpillar (CAT)	106
3.16	When Don't the Models Work?	109
3.17	Summary	113
	Exercises	113
4	Valuation Based on the Consolidated Statement of Cash Flows	117
4.1	Overview	117
4.2	Free Cash Flow (FCF): Measuring the Cash Produced by the Business	119
4.3	A Simple Example	121
4.4	Merck: Reverse Engineering the Market Value	124
4.5	Summary	126
	Exercise	126

5	Pro Forma Financial Statement Modeling	127
	5.1 Overview	127
	5.2 How Financial Models Work: Theory and an Initial Example	127
	5.3 Free Cash Flow (FCF): Measuring the Cash Produced by the Business	136
	5.4 Using the Free Cash Flow (FCF) to Value the Firm and Its Equity	138
	5.5 Some Notes on the Valuation Procedure	140
	5.6 Alternative Modeling of Fixed Assets	142
	5.7 Sensitivity Analysis	144
	5.8 Debt as a Plug	145
	5.9 Incorporating a Target Debt/Equity Ratio into a Pro Forma	148
	5.10 Project Finance: Debt Repayment Schedules	150
	5.11 Calculating the Return on Equity	153
	5.12 Tax Loss Carryforwards	155
	5.13 Summary	157
	Exercises	157
6	Building a Pro Forma Model: The Case of Caterpillar	161
	6.1 Overview	161
	6.2 Caterpillar's Financial Statements, 2007–2011	162
	6.3 Analyzing the Financial Statements	166
	6.4 A Model for Caterpillar	176
	6.5 Using the Model to Value Caterpillar	177
	6.6 Summary	178
7	Financial Analysis of Leasing	179
	7.1 Overview	179
	7.2 A Simple but Misleading Example	179
	7.3 Leasing and Firm Financing—The Equivalent-Loan Method	181
	7.4 The Lessor's Problem: Calculating the Highest Acceptable Lease Rental	184
	7.5 Asset Residual Value and Other Considerations	187
	7.6 Leveraged Leasing	189
	7.7 A Leveraged Lease Example	190
	7.8 Summary	193
	Exercises	193

II	PORTFOLIO MODELS	195
8	Portfolio Models—Introduction	197
	8.1 Overview	197
	8.2 Computing Returns for Apple (AAPL) and Google (GOOG)	197
	8.3 Calculating Portfolio Means and Variances	202
	8.4 Portfolio Mean and Variance—Case of N Assets	205
	8.5 Envelope Portfolios	210
	8.6 Summary	213
	Exercises	213
	Appendix 8.1: Adjusting for Dividends	215
	Appendix 8.2: Continuously Compounded Versus Geometric Returns	218
9	Calculating Efficient Portfolios	221
	9.1 Overview	221
	9.2 Some Preliminary Definitions and Notation	221
	9.3 Five Propositions on Efficient Portfolios and the CAPM	223
	9.4 Calculating the Efficient Frontier: An Example	227
	9.5 Finding Efficient Portfolios in One Step	234
	9.6 Three Notes on the Optimization Procedure	236
	9.7 Finding the Market Portfolio: The Capital Market Line (CML)	239
	9.8 Testing the SML—Implementing Propositions 3–5	242
	9.9 Summary	245
	Exercises	246
	Mathematical Appendix	248
10	Calculating the Variance-Covariance Matrix	251
	10.1 Overview	251
	10.2 Computing the Sample Variance-Covariance Matrix	251
	10.3 The Correlation Matrix	256
	10.4 Computing the Global Minimum Variance Portfolio (GMVP)	259
	10.5 Four Alternatives to the Sample Variance-Covariance Matrix	261
	10.6 Alternatives to the Sample Variance-Covariance: The Single-Index Model (SIM)	262
	10.7 Alternatives to the Sample Variance-Covariance: Constant Correlation	264

10.8	Alternatives to the Sample Variance-Covariance: Shrinkage Methods	266
10.9	Using Option Information to Compute the Variance Matrix	268
10.10	Which Method to Compute the Variance-Covariance Matrix?	271
10.11	Summary	272
	Exercises	272
11	Estimating Betas and the Security Market Line	273
11.1	Overview	273
11.2	Testing the SML	276
11.3	Did We Learn Something?	280
11.4	The Non-Efficiency of the “Market Portfolio”	283
11.5	So What’s the Real Market Portfolio? How Can We Test the CAPM?	285
11.6	Using Excess Returns	286
11.7	Summary: Does the CAPM Have Any Uses?	288
	Exercises	288
12	Efficient Portfolios Without Short Sales	291
12.1	Overview	291
12.2	A Numerical Example	292
12.3	The Efficient Frontier with Short-Sale Restrictions	298
12.4	A VBA Program for the Efficient Frontier Without Short Sales	299
12.5	Other Position Restrictions	302
12.6	Summary	303
	Exercise	303
13	The Black-Litterman Approach to Portfolio Optimization	305
13.1	Overview	305
13.2	A Naive Problem	307
13.3	Black and Litterman’s Solution to the Optimization Problem	313
13.4	BL Step 1: What Does the Market Think?	313
13.5	BL Step 2: Introducing Opinions—What Does Joanna Think?	316
13.6	Using Black-Litterman for International Asset Allocation	324
13.7	Summary	328
	Exercises	329

14	Event Studies	331
	14.1 Overview	331
	14.2 Outline of an Event Study	331
	14.3 An Initial Event Study: Procter & Gamble Buys Gillette	335
	14.4 A Fuller Event Study: Impact of Earnings Announcements on Stock Prices	342
	14.5 Using a Two-Factor Model of Returns for an Event Study	350
	14.6 Using Excel's Offset Function to Locate a Regression in a Data Set	355
	14.7 Summary	357
III	VALUATION OF OPTIONS	359
15	Introduction to Options	361
	15.1 Overview	361
	15.2 Basic Option Definitions and Terminology	361
	15.3 Some Examples	364
	15.4 Option Payoff and Profit Patterns	365
	15.5 Option Strategies: Payoffs from Portfolios of Options and Stocks	370
	15.6 Option Arbitrage Propositions	372
	15.7 Summary	379
	Exercises	380
16	The Binomial Option Pricing Model	383
	16.1 Overview	383
	16.2 Two-Date Binomial Pricing	383
	16.3 State Prices	385
	16.4 The Multi-Period Binomial Model	389
	16.5 Pricing American Options Using the Binomial Pricing Model	395
	16.6 Programming the Binomial Option Pricing Model in VBA	398
	16.7 Convergence of Binomial Pricing to the Black-Scholes Price	404
	16.8 Using the Binomial Model to Price Employee Stock Options	408
	16.9 Using the Binomial Model to Price Non-Standard Options: An Example	417
	16.10 Summary	419
	Exercises	419

17	The Black-Scholes Model	425
	17.1 Overview	425
	17.2 The Black-Scholes Model	425
	17.3 Using VBA to Define a Black-Scholes Pricing Function	427
	17.4 Calculating the Volatility	430
	17.5 A VBA Function to Find the Implied Volatility	434
	17.6 Dividend Adjustments to the Black-Scholes	437
	17.7 Using the Black-Scholes Formula to Price Structured Securities	441
	17.8 Bang for the Buck with Options	457
	17.9 The Black (1976) Model for Bond Option Valuation	459
	17.10 Summary	462
	Exercises	462
18	Option Greeks	467
	18.1 Overview	467
	18.2 Defining and Computing the Greeks	468
	18.3 Delta Hedging a Call	474
	18.4 Hedging a Collar	476
	18.5 Summary	485
	Exercises	486
	Appendix: VBA for Greeks	486
19	Real Options	493
	19.1 Overview	493
	19.2 A Simple Example of the Option to Expand	494
	19.3 The Abandonment Option	497
	19.4 Valuing the Abandonment Option as a Series of Puts	503
	19.5 Valuing a Biotechnology Project	505
	19.6 Summary	511
	Exercises	512
IV	VALUING BONDS	515
20	Duration	517
	20.1 Overview	517
	20.2 Two Examples	517

20.3	What Does Duration Mean?	520
20.4	Duration Patterns	524
20.5	The Duration of a Bond with Uneven Payments	525
20.6	Non-Flat Term Structures and Duration	533
20.7	Summary	536
	Exercises	536
21	Immunization Strategies	539
21.1	Overview	539
21.2	A Basic Simple Model of Immunization	539
21.3	A Numerical Example	541
21.4	Convexity: A Continuation of Our Immunization Experiment	545
21.5	Building a Better Mousetrap	547
21.6	Summary	551
	Exercises	551
22	Modeling the Term Structure	553
22.1	Overview	553
22.2	Basic Example	553
22.3	Several Bonds with the Same Maturity	558
22.4	Fitting a Functional Form to the Term Structure	562
22.5	The Properties of the Nelson-Siegel Term Structure	566
22.6	Term Structure for Treasury Notes	569
22.7	An Additional Computational Improvement	571
22.8	Nelson-Siegel-Svensson Model	573
22.9	Summary	574
	Appendix: VBA Functions Used in This Chapter	575
23	Calculating Default-Adjusted Expected Bond Returns	579
23.1	Overview	579
23.2	Calculating the Expected Return in a One-Period Framework	581
23.3	Calculating the Bond Expected Return in a Multi-Period Framework	582
23.4	A Numerical Example	587
23.5	Experimenting with the Example	589
23.6	Computing the Bond Expected Return for an Actual Bond	591

	23.7	Semiannual Transition Matrices	596
	23.8	Computing Bond Beta	599
	23.9	Summary	602
		Exercises	603
V		MONTE CARLO METHODS	605
24		Generating and Using Random Numbers	607
	24.1	Overview	607
	24.2	Rand() and Rnd: The Excel and VBA Random-Number Generators	608
	24.3	Testing Random-Number Generators	611
	24.4	Generating Normally Distributed Random Numbers	617
	24.5	Norm.Inv: Another Way to Generate Normal Deviates	628
	24.6	Generating Correlated Random Numbers	630
	24.7	What's Our Interest in Correlation? A Small Case	635
	24.8	Multiple Random Variables with Correlation: The Cholesky Decomposition	638
	24.9	Multivariate Normal with Non-Zero Means	646
	24.10	Multivariate Uniform Simulations	648
	24.11	Summary	651
		Exercises	651
25		An Introduction to Monte Carlo Methods	655
	25.1	Overview	655
	25.2	Computing π Using Monte Carlo	655
	25.3	Writing a VBA Program	661
	25.4	Another Monte Carlo Problem: Investment and Retirement	663
	25.5	A Monte Carlo Simulation of the Investment Problem	667
	25.6	Summary	671
		Exercises	671
26		Simulating Stock Prices	675
	26.1	Overview	675
	26.2	What Do Stock Prices Look Like?	676
	26.3	Lognormal Price Distributions and Geometric Diffusions	681
	26.4	What Does the Lognormal Distribution Look Like?	684

	26.5	Simulating Lognormal Price Paths	688
	26.6	Technical Analysis	692
	26.7	Calculating the Parameters of the Lognormal Distribution from Stock Prices	694
	26.8	Summary	696
		Exercises	696
27		Monte Carlo Simulations for Investments	699
	27.1	Overview	699
	27.2	Simulating Price and Returns for a Single Stock	699
	27.3	Portfolio of Two Stocks	702
	27.4	Adding a Risk-Free Asset	706
	27.5	Multiple Stock Portfolios	708
	27.6	Simulating Savings for Pensions	710
	27.7	Beta and Return	715
	27.8	Summary	720
		Exercises	720
28		Value at Risk (VaR)	723
	28.1	Overview	723
	28.2	A Really Simple Example	723
	28.3	Defining Quantiles in Excel	725
	28.4	A Three-Asset Problem: The Importance of the Variance-Covariance Matrix	728
	28.5	Simulating Data: Bootstrapping	730
		Appendix: How to Bootstrap: Making a Bingo Card in Excel	736
29		Simulating Options and Option Strategies	745
	29.1	Overview	745
	29.2	Imperfect but Cashless Replication of a Call Option	747
	29.3	Simulating Portfolio Insurance	750
	29.4	Some Properties of Portfolio Insurance	758
	29.5	Digression: Insuring Total Portfolio Returns	759
	29.6	Simulating a Butterfly	765
	29.7	Summary	771
		Exercises	772

30	Using Monte Carlo Methods for Option Pricing	775
	30.1 Overview	775
	30.2 Pricing a Plain-Vanilla Call Using Monte Carlo Methods	776
	30.3 State Prices, Probabilities, and Risk Neutrality	780
	30.4 Pricing a Call Using the Binomial Monte Carlo Model	782
	30.5 Monte Carlo Plain-Vanilla Call Pricing Converges to Black-Scholes	786
	30.6 Pricing Asian Options	794
	30.7 Pricing Asian Options with a VBA Program	802
	30.8 Pricing Barrier Options with Monte Carlo	807
	30.9 Using VBA and Monte Carlo to Price a Barrier Option	811
	30.10 Summary	817
	Exercises	817
VI	EXCEL TECHNIQUES	821
31	Data Tables	823
	31.1 Overview	823
	31.2 An Example	823
	31.3 Setting Up a One-Dimensional Data Table	824
	31.4 Building a Two-Dimensional Data Table	826
	31.5 An Aesthetic Note: Hiding the Formula Cells	827
	31.6 Excel Data Tables Are Arrays	828
	31.7 Data Tables on Blank Cells (Advanced)	829
	31.8 Data Tables Can Stop Your Computer	835
	Exercises	836
32	Matrices	839
	32.1 Overview	839
	32.2 Matrix Operations	840
	32.3 Matrix Inverses	843
	32.4 Solving Systems of Simultaneous Linear Equations	845
	32.5 Some Homemade Matrix Functions	846
	Exercises	851

33	Excel Functions	855
	33.1 Overview	855
	33.2 Financial Functions	855
	33.3 Dates and Date Functions	863
	33.4 The Functions XIRR, XNPV	869
	33.5 Statistical Functions	875
	33.6 Regressions with Excel	879
	33.7 Conditional Functions	889
	33.8 Large and Rank, Percentile, and PercentRank	890
	33.9 Count, CountA, CountIf, CountIfs, AverageIf, AverageIfs	891
	33.10 Boolean Functions	894
	33.11 Offset	896
34	Array Functions	899
	34.1 Overview	899
	34.2 Some Built-In Excel Array Functions	899
	34.3 Homemade Array Functions	904
	34.4 Array Formulas with Matrices	907
	Exercises	911
35	Some Excel Hints	913
	35.1 Overview	913
	35.2 Fast Copy: Filling in Data Next to Filled-In Column	913
	35.3 Filling Cells with a Series	915
	35.4 Multi-Line Cells	916
	35.5 Multi-Line Cells with Text Formulas	917
	35.6 Writing on Multiple Spreadsheets	918
	35.7 Moving Multiple Sheets of an Excel Notebook	919
	35.8 Text Functions in Excel	920
	35.9 Chart Titles That Update	920
	35.10 Putting Greek Symbols in Cells	924
	35.11 Superscripts and Subscripts	925
	35.12 Named Cells	926
	35.13 Hiding Cells (in Data Tables and Other Places)	928
	35.14 Formula Auditing	930
	35.15 Formatting Millions as Thousands	932
	35.16 Excel's Personal Notebook: Automating Frequent Procedures	934

VII	VISUAL BASIC FOR APPLICATIONS (VBA)	943
36	User-Defined Functions with VBA	945
	36.1 Overview	945
	36.2 Using the VBA Editor to Build a User-Defined Function	945
	36.3 Providing Help for User-Defined Functions in the Function Wizard	955
	36.4 Saving Excel Workbook with VBA Content	958
	36.5 Fixing Mistakes in VBA	960
	36.6 Conditional Execution: Using If Statements in VBA Functions	963
	36.7 The Boolean and Comparison Operators	967
	36.8 Loops	970
	36.9 Using Excel Functions in VBA	977
	36.10 Using User-Defined Functions in User-Defined Functions Exercises	979
	Appendix: Cell Errors in Excel and VBA	981
37	Variables and Arrays	986
	37.1 Overview	989
	37.2 Defining Function Variables	989
	37.3 Arrays and Excel Ranges	992
	37.4 Simple VBA Arrays	995
	37.5 Multidimensional Arrays	1005
	37.6 Dynamic Arrays and the ReDim Statement	1007
	37.7 Array Assignment	1009
	37.8 Variants Containing an Array	1011
	37.9 Arrays as Parameters to Functions	1012
	37.10 Using Types	1015
	37.11 Summary	1016
	Exercises	1017
38	Subroutines and User Interaction	1023
	38.1 Overview	1023
	38.2 Subroutines	1023
	38.3 User Interaction	1030
	38.4 Using Subroutines to Change the Excel Workbook	1033

	38.5 Modules	1036
	38.6 Summary	1040
	Exercises	1040
39	Objects and Add-Ins	1047
	39.1 Overview	1047
	39.2 Introduction to Worksheet Objects	1047
	39.3 The Range Object	1049
	39.4 The With Statement	1053
	39.5 Collections	1055
	39.6 Names	1061
	39.7 Add-Ins and Integration	1064
	39.8 Summary	1068
	Exercises	1068
	Selected References	1073
	Index	1085

Preface

The three previous editions of *Financial Modeling* have received a gratifyingly positive response from readers. The combination of a “cookbook,” mixing explanation and implementation using Excel, has fulfilled a need in both the academic and the practitioner markets from readers who realize that the implementation of the finance basics typically studied in an introductory finance course requires another, more heavily computational and implementational approach. Excel, the most widely used computational tool in finance, is a natural vehicle for deepening our understanding of the materials.

In this fourth edition of *Financial Modeling*, I have added a section (Chapters 24–30) on Monte Carlo methods. The intention is to add a focus on the simulation of financial models. I have become convinced that a statistical understanding of modeling (“What is the mean and sigma of the portfolio return?”) understates the impact of the uncertainty. Only by simulating the models and the return processes can we get a good feel for the dimensions of the uncertainty.

With the added section on Monte Carlo, *Financial Modeling* now consists of seven sections. Each of the first five sections of the book relates to a specific area of finance. These sections are independent of each other, though the reader should realize that they all assume some familiarity with the finance area—*Financial Modeling* is not an introductory text. Section I (Chapters 1–7) deals with corporate finance topics; Section II (Chapters 8–14) with portfolio models; Section III (Chapters 15–19) with option models; and Section IV (Chapters 20–23) with bond-related topics. Section V, as discussed above, introduces the reader to Monte Carlo methods in finance.

The last two sections of *Financial Modeling* are technical in nature. Section VI (Chapters 31–35) relates to various Excel topics which are used throughout the book. Chapters in Section VI can be read and accessed as necessary. Section VII (Chapters 36–39) deals with Excel’s programming language, Visual Basic for Applications (VBA). VBA is used throughout *Financial Modeling* to create functions and routines which make life easier, but it is never intrusive—in principle the reader can understand the materials in all of the other chapters of *Financial Modeling* without needing the VBA chapters.

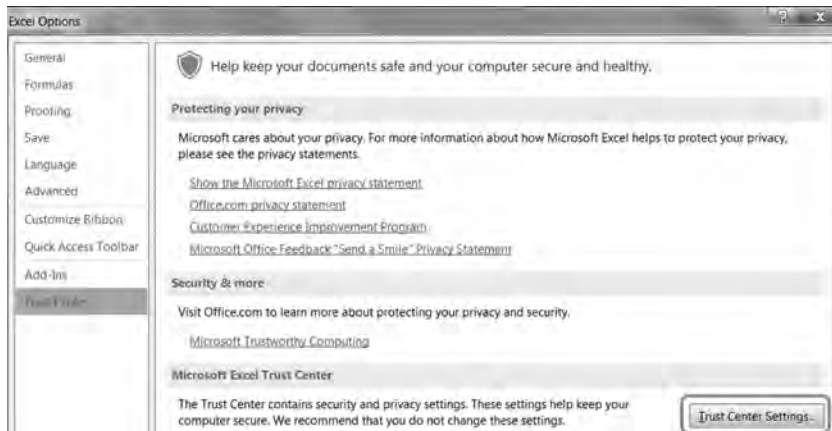
New Materials and Updates

This edition of *Financial Modeling* contains much new and updated material. We have already mentioned the new section on Monte Carlo methods. Also new are two chapters on valuation (Chapters 2 and 4) and a chapter on term structure modeling (Chapter 22). Much of the material has been tweaked

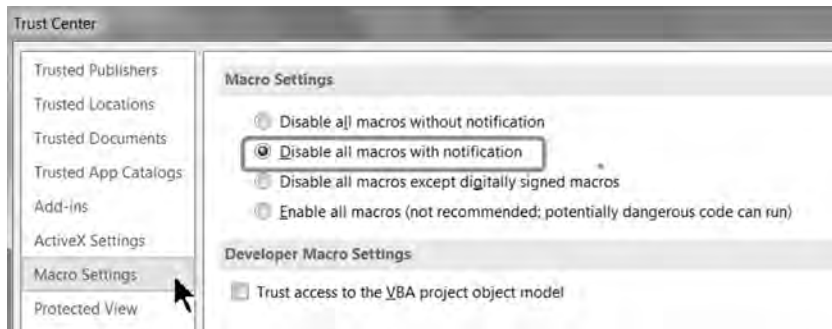
and improved. For example, the discussion of Excel financial functions now includes a discussion of XIRR and XNPV, including a fix for the bugs in these functions.

Getformula

The Excel files with this edition include a function called **Getformula** that enables the user to track cell contents. **Getformula** is discussed in Chapter 0 and also on a file on the disk that is included with *Financial Modeling*. To allow **Getformula** to work, go to **File|Options|Trust Center**:



In the **Trust Center** settings, I recommend the following setting:



If you have done this, then when opening an Excel notebook for the first time, you will be confronted by the following warning:



For notebooks that come with this book, you can safely click **Enable Content**, which enables the formulas on the notebook.

Excel Versions

In the examples throughout the book I have used Excel 2013. To the best of my knowledge, all of the spreadsheets work in Excel versions 2003, 2007, 2010, and 2011 (for Mac), although some minor and obvious adaptations by the reader may be called for.

Files for the Fourth Edition

Purchasers of *Financial Modeling* get access to all the Excel files for the chapters and exercises.

Using *Financial Modeling* in a University Course

Financial Modeling has become the book of choice in many advanced finance classes that stress the combination of modeling/Excel skills and a deeper understanding of the underlying financial models. The *Financial Modeling*-based courses are often a third- or fourth-year undergraduate or second-year MBA course. The courses are very different and include much instructor-specific input, but they seem to have a few general features in common:

- A typical course starts with two or three classes which stress the Excel skills needed for financial modeling. Often these courses are held in a computer lab. Though almost all business school students know Excel, they often do not know the finesses of data tables (Chapter 31), some of the basic financial functions (Chapters 1 and 33), and array functions (Chapter 34).
- Most one-semester courses then cover at most one of the *Financial Modeling* sections. If we assume that in a typical university course, covering one chapter per week is an upper limit (and many chapters will require two weeks), then a typical course might concentrate on either corporate finance (Chapters 1–7),

portfolio models (Chapters 8–14), or options (Chapters 15–19). At a stretch, the instructor could perhaps throw in the shorter bond section (Chapters 20–23).

- I suggest that after the initial classes in a computer lab, the instructor move to a regular classroom. This enables the classroom emphasis to be on discussions of theory and implementation, with student homework concentrating on actual spreadsheets.

A major problem with a computer-based course is how to structure the final examination. Two solutions seem to work well. One alternative is to have students (whether alone or in teams) submit a final project; examples might be a corporate valuation if the course is based on Section I of the book, an event study for Section II, an option-based project for Section III, or the computation of a bond-expected return if the emphasis is on Section IV. A second alternative is to have students submit, by e-mail, a spreadsheet-based examination with severe time limits. One instructor using this book sends his class the final exam (a compendium of spreadsheet problems) at 9 in the morning and requires an e-mail with a spreadsheet answer by noon.

Acknowledgments

I thank a number of people who have made materially significant comments to this edition:

Meni Abudy, Zvika Afik, Javierma Bedoya, Lisa Bergé, Elizabeth Caulk, Sharon Garyn-Tal, Victor Lampe, Jongdoo Lee, Erez Levy, Warren Miller, Tal Mofkadi, Roger Myerson, Siddhartha Sarkar, Maxim Sharov, Permjit Singh, Sondre Aarseth Skjerven, Alexander Suhov, Kien-Quoc Van Pham, Chao Wang, Tim Wu.

Finally, I would like to thank: my editor John Covell of MIT Press, Ellen Faran, the Director of MIT Press, and Nancy Benjamin and her editorial team at Books By Design. They have all been unfailingly helpful and patient.

Disclaimer

The materials in this book are intended for instructional and educational purposes only, to illustrate situations similar to those encountered in the real world. They may not apply directly to real-world situations. The author and MIT Press disclaim any responsibility for the consequences of implementation.

From the Preface to the Third Edition

The two previous editions of *Financial Modeling* have received a gratifyingly positive response from readers. The combination of a “cookbook,” mixing explanation and implementation using Excel has fulfilled a need in both the academic and the practitioner markets from readers who realize that the implementation of the finance basics typically studied in an introductory finance course requires another, more heavily computational and implementational, approach. Excel, the most widely used computational tool in finance, is a natural vehicle for deepening our understanding of the materials.

Acknowledgments

I want to start by thanking a group of wonderful editors: John Covell, Nancy Lombardi, Elizabeth Murry, Ellen Pope, and Peter Reinhart. My next thanks go to a dedicated group of colleagues who read the typescripts for *Financial Modeling*: Michael Chau, Jaksa Cvitanic, Arindam Bandopadhyaya, Richard Harris, Aurele Hougbedji, Iordanis Karagiannidis, Yvan Lengwiler, Nejat Seyhun, Gökçe Soydemir, David Y. Suk.

Many of the changes in this edition of *Financial Modeling* are due to the comments of readers, who have been assiduous in offering suggestions and improvements in the book. I follow a tradition started with the first two editions of *Financial Modeling* by acknowledging those readers whose comments have been incorporated into this edition:

Meni Abudy, Zvika Afik, Gordon Alexander, Apostol Bakalov, Naomi Belfer, David Biere, Vitaliy Bilyk, Oded Braverman, Roeland Brinkers, Craig Brody, Salvio Cardozo, Sharad Chaudhary, Israel Dac, Jeremy Darhansoff, Toon de Bakker, Govindvyas Dharwada, Davey Disatnik, Kevin P. Dowd, Brice Dupoyet, Cederik Engel, Orit Eshel, Yaara Geyra, Rana P. Ghosh, Bjarne Jensen, Marek Jochec, Milton Joseph, Erez Kamer, Saggi Katz, Emir Kiamilev, Brennan Lansing, Paul Ledin, Paul Legerer, Quinn Lewis, David Martin, Tom McCurdy, Tsahi Melamed, Tal Mofkadi, Geoffrey Morrisett, Sandip Mukherji, Max Nokhrin, Michael Oczkowski, David Pedersen, Mikael Petitjean, Georgio Questo, Alex Riahi, Arad Rostampour, Joseph Rubin, Andres Rubio, Ofir Shatz, Natalia Simakina, Ashutosh Singh, Permjit Singh, Gerald Strever, Shavkat Sultanbekov, Ilya Talman, Mel Tukman, Daniel Vainder, Guy Vishnia, Torben Voetmann, Chao Wang, James Ward, Roberto Wessels, Geva Yaniv, Richard Yeh, and Werner Zitzman.

Finally, I want to thank my very patient wife, Terry, who has maintained her own and my equilibrium through two books and a business school deanship in the past five years.

From the Preface to the Second Edition

The purpose of this book remains to provide a “cookbook” for implementing common financial models in Excel. This edition has been expanded by six additional chapters, covering financial calculations, cost of capital, value at risk (VaR), real options, early exercise boundaries, and term-structure modeling. There is also an additional technical chapter containing a potpourri of Excel hints.

I am indebted to a number of people (in addition to those mentioned in the previous preface) for help and suggestions: Andrew A. Adamovich, Alejandro Sanchez Arevalo, Yoni Aziz, Thierry Berger-Helmchen, Roman Weissman Bermann, Michael Giacomo Bertolino, John Bollinger, Enrico Camerini, Manuel Carrera, Roy Carson, John Carson, Lydia Cassorla, Philippe Charlier, Michael J. Clarke, Alvaro Cobo, Beni Daniel, Ismail Dawood, Ian Dickson, Moacyr Dutra, Hector Tassinari Eldridge, Shlomy Elias, Peng Eng, Jon Fantell, Erik Ferning, Raz Gilad, Nir Gluzman, Michael Gofman, Doron Greenberg, Phil Hamilton, Morten Helbak, Hitoshi Hibino, Foo Siat Hong, Marek Jochech, Russell W. Judson, Tiffani Kaliko, Boris Karasik, Rick Labs, Allen Lee, Paul Legerer, Guoli Li, Moti Marcus, Gershon Mensher, Tal Mofkadi, Stephen O’Neil, Steven Ong, Oren Ossad, Jackie Rosner, Steve Rubin, Dvir Sabah, Ori Salinger, Meir Shahar, Roger Shelor, David Siu, Maja Sliwinski, Bob Taggart, Maurry Tamarkin, Mun Hon Tham, Efrat Tolkowsky, Mel Tukman, Sandra van Balen, Michael Verhofen, Lia Wang, Roberto Wessels, Ethan Weyand, Ubbo Wiersema, Weiqin Xie, Ke Yang, Ken Yook, George Yuan, Khurshid Zaynutdinov, Ehud Ziegelman, and Eric Zivot. I also want to thank my editors, who again have been a great help: Nancy Lombardi, Peter Reinhart, Victoria Richardson, and Terry Vaughn.

From the Preface to the First Edition

Like its predecessor *Numerical Techniques in Finance*, the aim of this book is to present some important financial models and to show how they can be solved numerically and/or simulated using Excel. In this sense this is a finance “cookbook;” like any cookbook, it gives recipes with a list of ingredients and instructions for making and baking. As any cook knows, a recipe is just a starting point; having followed the recipe a number of times, you can think of your own variations and make the results suit your tastes and needs.

Financial Modeling covers standard financial models in the areas of corporate finance, financial statement simulation, portfolio problems, options, portfolio insurance, duration, and immunization. The aim in each case has been to explain clearly and concisely the implementation of the models using Excel. Very little theory is offered except where necessary to understand the numerical implementations.

While Excel is often not the tool to use for high-level, industrial-strength calculations (portfolios are an example), it is an excellent tool for understanding the computational intricacies involved in financial modeling. It is often the case that the fullest understanding of the models comes by calculating them, and Excel is one of the most accessible and powerful tools available for this purpose.

Along the way a lot of students, colleagues, and friends (these are nonexclusive categories) have helped me with advice and comments. In particular I would like to thank Olivier Blechner, Miryam Brand, Elizabeth Caulk, John Caulk, Benjamin Czaczkes, John Ferrari, John P. Flagler, Dan Fylstra, Kuni-hiko Higashi, Julia Hynes, Don Keim, Anthony Kim, Ken Kunimoto, Rick Labs, Adrian Lawson, Philippe Nore, Isidro Sanchez Alvarez, Nir Sharabi, Edwin Strayer, Robert Taggart, Mark Thaler, Terry Vaughn, and Xiaoge Zhou.

Finally, my thanks go to a wonderful set of editors: Nancy Lombardi, Peter Reinhart, Victoria Richardson, and Terry Vaughn.

O

Before All Else

0.1 Data Tables

Financial Modeling makes extensive use of data tables. I advise readers of the book to first make sure that they understand data tables (read Chapter 31, sections 1–5). Data tables are absolutely critical in the sensitivity analysis that is part of most financial models. They are a little bit complicated, but an invaluable addition to the modeling arsenal of the financial modeler.

In the remainder of this short chapter, I discuss **Getformula**.

0.2 What Is Getformula?

The Excel notebooks in *Financial Modeling*, fourth edition, contain a function called **Getformula** that aids in annotating your spreadsheets. In the example below, cell C5 shows the formula contained in cell B5; the formula in question computes the annual repayment of a loan of 165,000 for 7 years at 8%. Cell C5 contains the function =**Getformula(B5)**.

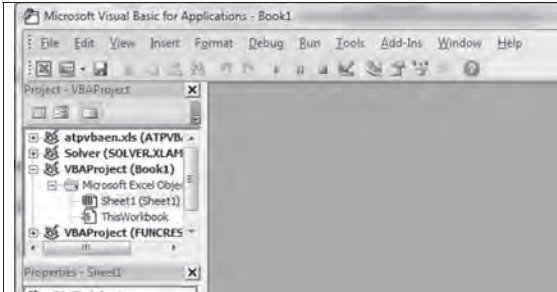
	A	B	C
2	Principal	165,000	
3	Interest	8%	
4	Term	7	<-- years
5	Annual payment	31,691.95	<-- =PMT(B3,B4,-B2)

In this short chapter, we describe how to add this formula to your Excel notebook. Mac users: This works only in Excel 2011.

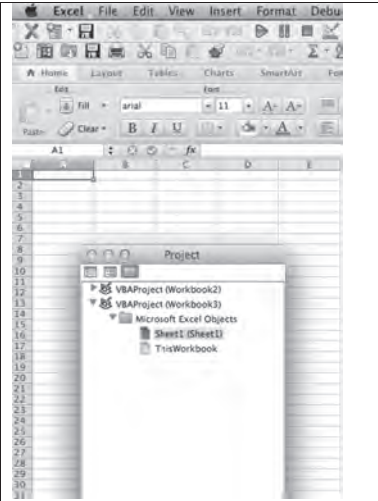
0.3 How to Put Getformula into Your Excel Notebook

1. Open the Excel workbook in which you want the formula to work.
2. Open the VBA editor:
 - On Windows computers: Press [Alt] + F11.
 - On Mac (Excel 2011): Choose **Tools|Macro|Visual Basic Editor**

3. This will open the VBA editor.

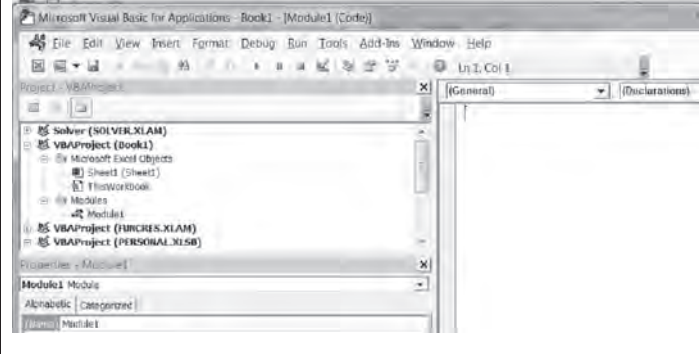
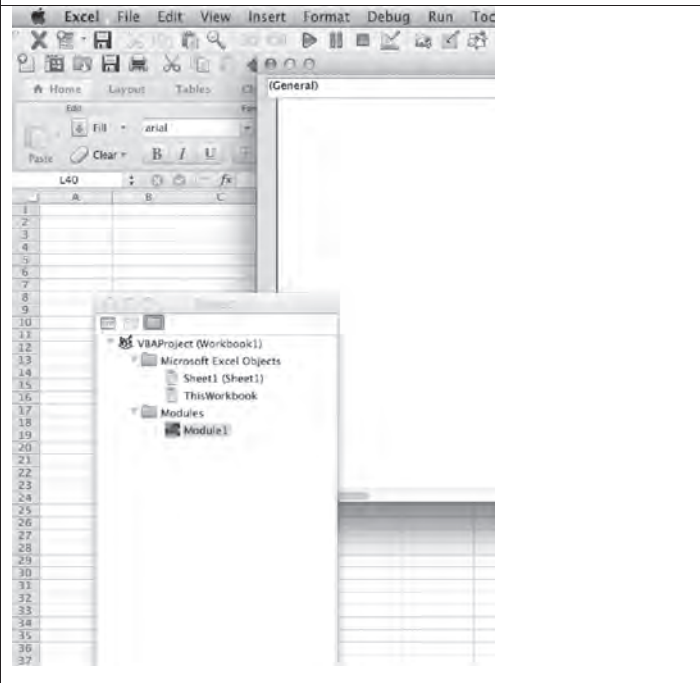


Windows screen



Mac screen

4. Select **Insert|Module** at the top of the screen.

	<p>← Windows screen</p>
	<p>← Mac screen</p>

5. Now insert the following text into the Module window (where it says **General**). Just copy/paste the text below.

```

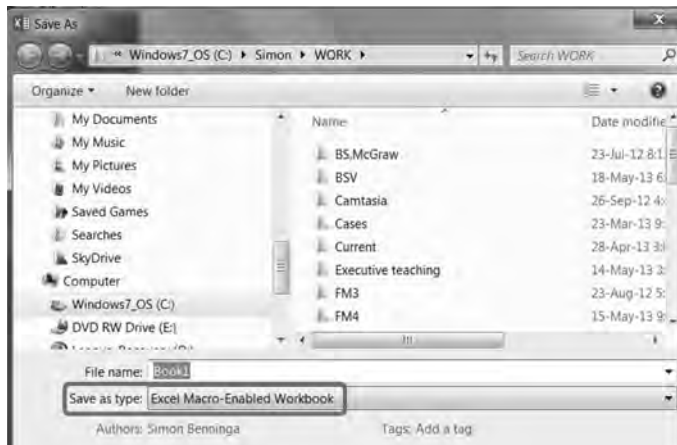
`8/5/2006 Thanks to Maja Sliwinski and
`Beni Czaczkes
Function getformula(r As Range) As String
    Application.Volatile
    If r.HasArray Then
        getformula = "<-- " & _
            " {" & r.FormulaArray & "}"
    Else
        getformula = "<-- " & _
            " " & r.FormulaArray
    End If
End Function

```

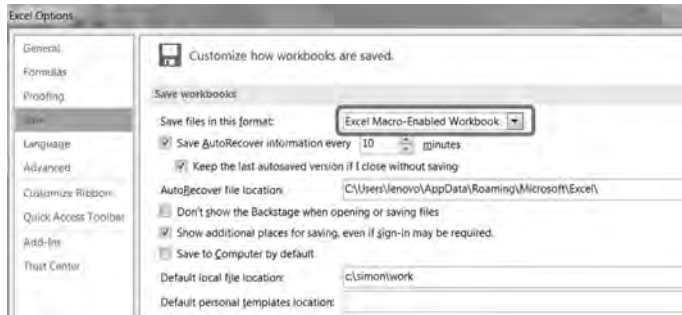
In Windows, close the VBA window (no need to save). On the Mac, just continue to work on the spreadsheet. The formula is now part of the spreadsheet and will be saved along with it.

0.4 Saving the Excel Workbook: Windows

To save the notebook with the **Getformula** macro in VBA, you will have to save it as a **Macro-enabled workbook**.

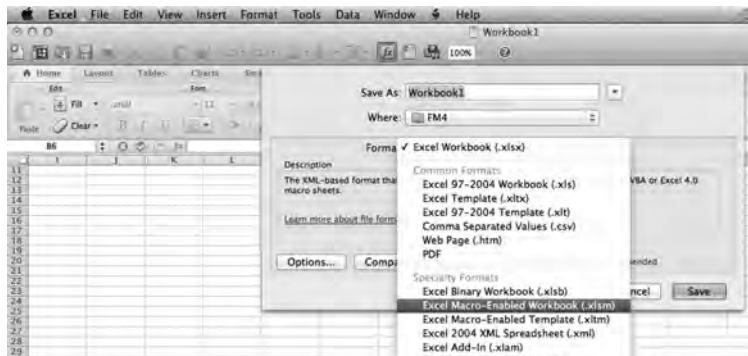


Macro-enabled workbooks have the extension .xlsm, whereas regular Excel workbooks have the extension .xlsx. Your users will never know the difference. We have changed our Excel settings (**File|Options|Save**) to make the Macro-enabled workbook our default:



0.5 Saving the Excel Workbook: Mac

The Mac screen for saving as a Macro-enabled workbook looks like this:



0.6 Do You Have to Put Getformula into Each Excel Workbook?

The short answer is “yes.” You could create an add-in to Excel (see Chapter 39) that contains **Getformula**, but this will make it more difficult for you to share your workbooks. We prefer to put **Getformula** in each new spreadsheet we create.

0.7 A Shortcut to Use Getformula

Once you have put **Getformula** into your Excel workbook, you will have to use it! Ninety percent of our uses of this function point to the cell to the left of the formula itself:

	A	B	C	D	E
1	QUESTION 2				
2	Interest rate	11%			
3					
4	Year	Asset1	Asset2	Asset3	
5	1	1,000	0	0	
6	2	1,000	0	0	
7	3	1,000	1,700	0	
8	4	1,000	1,700	0	
9	5	1,000	1,700	3,000	
10	6	1,000	0	4,000	
11	7	1,000	0	5,000	
12					
13	Value	4,712	3,372	6,327	=NPV(\$B\$2,D5:D11)
14					
15					

We’ve put a short macro into our **Personal workbook** that automates this procedure. The remainder of this section describes how to automate the **Getformula** procedure.

Automating the Procedure

We want to automate this procedure of putting **Getformula** into a cell:

- Turn it into a macro.
- Attach a key sequence (in our case, [Ctrl] + t) to the macro.
- Make the macro and key sequence available in your Excel spreadsheets.

We will save the macro to our **Personal.xlsb** file. This file activates each time you start Excel. It's yours only—other readers of your spreadsheets won't see it. Below we describe the steps, for both Windows and the Mac.

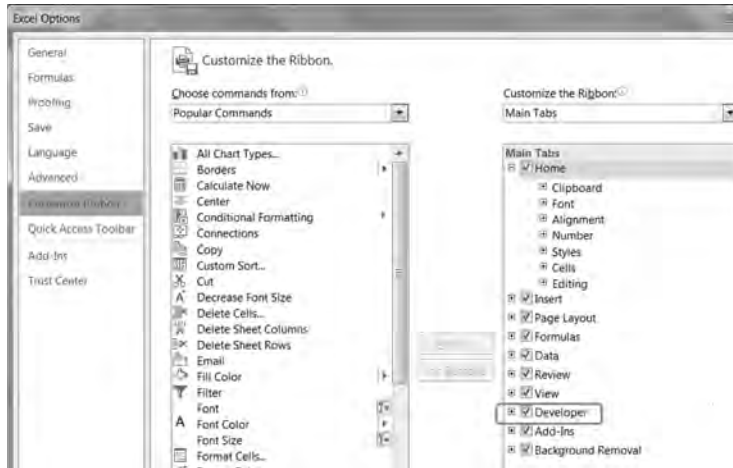
0.8 Recording Getformula: The Windows Case

Here are the steps to recording the macro in Windows:

- Activate the **Developer** tab on the menu bar.
- Use **Record Macro** to save a macro as a personal notebook.

Activate the Developer Tab

Go to **File|Options|Customize Ribbon** and activate the **Developer** tab as shown below:



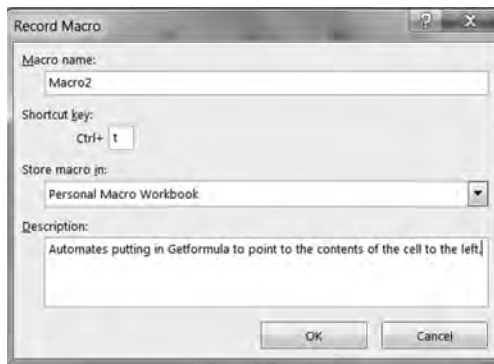
Use Record Macro

The Developer tab allows you to record a macro and save it as part of the Personal.xlsb notebook. We will illustrate with the copy as picture feature.

1. Open a blank Excel notebook and click on the **Developer** tab and then on **Record Macro**:



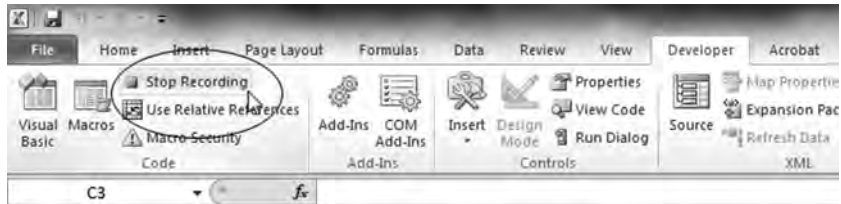
Excel will ask for details of the recording. Here's what I wrote. We will save this as a **Personal Macro Workbook** and then use the shortcut [Ctrl] + t:



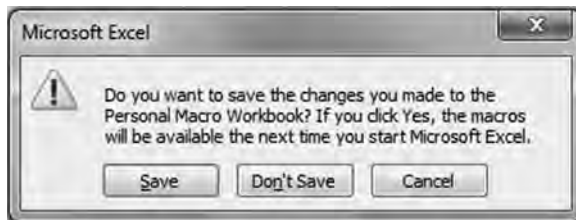
2. Now go to your spreadsheet and use **Getformula**, pointing to the cell to the left of where you want **Getformula** to appear. In the spreadsheet below, we have typed =Getformula(A3) into cell B4:

	A	B
1	QUESTION 1	
2	2	
3	3	
4	5 <--	=SUM(A2:A3)
5		

3. Go back to the **Developer** tab and stop the recording:



4. Close down Excel. Excel will ask you if you want to save the Personal workbook. The answer is, of course, positive:



This creates the following file (“simon benninga” is of course my user name on my computer—you will substitute your user name).

```
C:\Users\simon benninga\AppData\Roaming\Microsoft\Excel\XLSTART\
PERSONAL.XLSB
```

Using the Macro

From now on, whenever you open a file on *your computer*, you can use [Ctrl] + t to copy a region as a picture. Cool!