

# FIN 4934 - FINANCIAL MODELING

Fall 2020

Fully Online, Asynchronous

---

## Instructor

Nick DeRobertis

## Email

derobertisna@ufl.edu

## Office Hours

TBD

---

## COURSE MATERIALS

### Course Website

<https://nickderobertis.github.io/fin-model-course>

- The course website includes all of the materials for the course, including the course schedule, lecture videos, lab exercises, practice problems, and projects.

### Textbook

Simon Benninga, "Financial Modeling," Fourth Edition, MIT Press.

- Textbook is only recommended. This book is very focused on Excel and we will be using both Excel and Python in the course. It is however useful as a reference for the Excel material and generally how to build the models.

### Prepare for Class

Have a computer with Microsoft Excel 2007 or newer ready. We will be installing Python on this machine as well.

---

## GENERAL INFORMATION

### Description

This course covers the full financial modelling workflow using both Excel and Python. I will try to teach you how to build a model in general, all the way from concept and data collection to the result and visualization, and how to complete the various steps in either Excel or Python. My goal is to give you tangible skills which would be applicable in a finance job, and more so than any other course you've ever taken. This course will be very challenging for the less technically inclined, but I will try to help everyone succeed.

### Prerequisites

#### Courses

FIN 3403 (Business Finance) is the minimum requirement. It is highly recommended that you have also taken FIN 4243 (Debt and Money Markets) and FIN 4504 (Equity and Capital Markets) as we will build models based on the concepts in these courses without a thorough treatment of them. We have a lot to cover already in the course and so the finance concepts are not the focus. I can recommend textbooks for any concepts which you haven't covered in prior courses, but then you will have a large learning curve.

## Technical Skills

Understanding of algebra is necessary to work with equations in the models. Introductory Excel capabilities are required. Any experience with Python is a plus. It will be assumed that most students have introductory to intermediate skills in Excel and no experience with Python. Ensure that you have the Excel skills mentioned here. If you have no Excel experience or some gaps in your experience but still want to take the course, take a look at the free Excel resources at the end of the syllabus and contact me if you have questions as you go through the resources.

- Algebra
- Excel
  - Data entry, editing
  - Number and cell formatting
  - How to work with formulas, and knowledge of common math formulas
  - Cell references - fixed and relative, when to use each and why
  - Managing worksheets and workbooks
  - Graphs, charts

## Class Structure

### A Typical Day

The class is structured in three parts: lecture videos, lecture review sessions, and interactive lab sessions. The lecture review sessions will be a group video call which also be recorded and provide an opportunity for students to ask general questions about the lecture videos. You will have weekly assignments that reinforce the content from the lectures for that week, and they are the focus of the interactive lab sessions. The lab sessions will also be group video calls where students are encouraged to work on the problems and ask me questions. You may need to spend time outside of the lab sessions completing the exercises as well. You are encouraged to discuss the lab exercises with your classmates, but everyone should complete the exercises on their own computer.

## Projects

There will be four or five projects in the course. The grading in this course is entirely project-based, between the projects and the lab exercises.

Projects are to be completed individually. Do not copy each other's code or workbooks. This includes copy/paste as well as manually following someone's specific steps. You can however discuss them with your classmates, so long as the final submission is entirely your own work.

## Questions?

There are four ways you may ask questions: in lecture review sessions, in lab sessions, by email, and by appointment. Before you ask a question, be sure that you have watched the relevant lecture(s), and if the question is directly asking a question clearly covered in the lecture I may only point you to that.

If you are not able to figure out the lab exercise in the time provided, I am happy to help you by email or appointment for a video call.

## Late Policy

I will accept late projects, **but I will subtract 10% from the grade for each day late**. Projects are **due by midnight on the due date** and must be submitted to Canvas by this time. If you turn it in five minutes after midnight, I will subtract 10%. Then each additional 10% subtraction happens at midnight on each following day. But **once we review the project as a class, you are no longer allowed to submit it**. This will usually be about a week after but not always.

For example, if a project was due Tuesday, and you submit it two minutes after midnight (Wednesday 12:02 AM), you will lose 10%. Then, starting 12:01 AM Thursday (Wednesday night), you are losing 20%. Then starting 12:01 Friday (Thursday night), you are losing 30%. But we review the project as a class that Thursday, you may only submit it up until that time on Thursday.

## How to be Successful in This Class

I highly recommend using some outside resources to learn some basic Python as it will allow you to focus more on the course content. I will be teaching from the basics, but this is not a programming class so we don't have the time to give the language a proper treatment. In the past, students which started early with external Python and Excel resources performed well and got more applicable knowledge out of the class.

Watching every lecture is important as the knowledge from prior lectures becomes essential to understand the later lectures. Beyond the lectures, we will have interactive lab sessions where you can ask me questions and actively get help with the course material as you work on it. For the projects, you will need to build upon (directly or indirectly) what you have put together in the lab sessions. Practice problems will also be provided, and it is encouraged that you complete them before the projects. Start the projects early as there is considerable work involved. Many students have said they have been the most challenging and rewarding projects they completed at UF.

If you feel lost at any time, please contact me and I can provide additional help, as the material will continue to build on itself. You cannot afford to feel lost and do nothing and hope it will get better, it will only get worse as the course progresses.

## Make-Up Policy

Projects may be turned in late for reduced credit but not after we have reviewed it. See Projects section above.

## GRADES

---

### Breakdown

Item	Grade Percentage
Lab Exercises	20%
Projects	80%

### Grading Scale

Grade	Grade Percentage
A	93 - 100%
A-	90 - 92%
B+	87 - 89%
B	83 - 86%
B-	80 - 82%
C+	77 - 79%
C	73 - 76%
C-	70 - 72%
D+	67 - 69%
D	63 - 66%
D-	60 - 62%
F	0 - 59%

### Rounding

I will strictly follow standard rounding rules to two decimal places, so 92.450% is the minimum grade for an A. 92.449% is considered an A-.

### Curve

I will strive for a B average in the class. If this requires boosting grades with a curve, this will be done after all the projects are submitted. I will not reduce grades with a curve even if grades are higher than a B average.

## MAIN TOPICS COVERED

---

Subject to change but in approximate order.

### Introduction to Financial Modeling

- Overview
- Basic technical skills and setup – Excel and Python
- Time value of money models
- Basic statistical tools
- Monte Carlo methods

## Corporate Valuation

- Capital budgeting
- Estimating beta
- Estimating market value of debt
- Weighted average cost of capital (WACC)
- Free cash flow (FCF)
- Pro forma financial statements
- Discounted cash flow (DCF) valuation

## RESOURCES

---

### Python

#### Interactive + Video Tutorials

- [Python from Scratch](#) (University of Waterloo)

#### Interactive Tutorials

- [Computer Science Circles](#) (University of Waterloo)
- [Learn Python 2](#) (codecademy) | This one teaches Python 2, which is outdated, but the main difference in the covered material is just that the print function doesn't use parentheses in Python 2. So if you do this one then just adjust how you call print for the class. There is also a Python 3 course but it requires the Pro plan.
- [learnpython.org](#) | This is more fast paced

#### Video Tutorials

- [YouTube Video Series](#) (CS Dojo)
- [YouTube Video Series](#) (Jayanam)

#### Text (Non-Interactive) Tutorials

- [Non-Programmer's Tutorial for Python 3](#) (Wikibooks) | Fairly detailed and approachable
- [Hands-on Python 3 Tutorial](#) () | Very detailed and a bit faster paced
- [The Python Guru](#) | Fast paced, this one would be good for those who already know programming but not Python

### Excel

#### Video Tutorials

- [Trump Excel](#)
- [Excel 2016 Tutorial](#) (GCF Global)
- [Excel Exposure](#) (Ben Currier)