

# Course Overview

---

Welcome to BioSystems Analytics (BE434/534)! In this course, you'll learn how to write programs in Python through games and short coding exercises. You're about to embark on an exciting journey to learn an important and in-demand skill set.

## Course prerequisites:

This course builds your Python skills from the ground up! No prior skills are required. Just lots of good-old fashioned practice...

## Course objectives:

You will learn about Python through simple coding examples. This will give you an understanding of

- Basic Python syntax (if, for, while, defining functions)
- Using common Python data structures (strings, lists, tuples, and dictionaries)
- Importing and using additional Python modules for testing code (pytest, flake8, mypy)
- Using data directly from files and directories
- Coding real-world examples

## Creating your development environment:

To get the most out of this course, I strongly recommend that you install all of the required software on your machine (see setup documents). You will be writing and testing Python code directly on your machine.

## How to pass the class:

Each week, we will cover one chapter in the text for the class [Tiny Python Projects](#) by Ken Youens-Clark. This text is available free of charge through the University of Arizona online catalog. Each week you will need to:

- Watch the short video lectures, providing an overview of the chapter
- Read the chapter
- Complete the practice exercises for the chapter (in "examples" in the class GitHub repository)
- Complete the practice quiz for the chapter (take multiple times to improve your score and learn)
- Complete the homework assignment for the chapter (in the class GitHub repository)
- Use the slack discussion forum to ask questions, get help, or go deeper

In the last two weeks of the semester, you will get the opportunity to complete a final project to use all of the wonderful skills you have acquired. Projects will vary at the graduate- or undergraduate-level.

## How deadlines work:

Each week, the week's course content will be made available on Monday at 9am, and all tasks must be complete by Sunday at 11:59pm. These deadlines help you organize your time, and keep you on track. Chapters build on one another, so you must complete each chapter before proceeding to the next. Working on each chapter, each week will get you the practice you need to learn the wonders of Python!

## Submitting homework

You'll be using an application called VS Code for writing and testing Python programs, and submitting these programs to a web-based GitHub repository (shared with me, your instructor). Don't worry! You will learn how to get everything set up on your computer this week to make you successful (see the setup documents).

## Getting and giving help

Here are a few ways you can give and get help:

Discussion forums: You can share information and ideas with your fellow students in the discussion forum on slack. Slack is also great place to find answers to questions you may have that have already been asked. If you're stuck on a concept, are struggling to solve a practice exercise, or you just want more information on a subject, the discussion forums are there to help you move forward.

Online: Throughout this course, we'll teach you how to code in Python, but this is a big topic! We will provide a ton of information through videos, reading, examples, and supplemental readings, but sometimes, you may need to look things up on your own (this is true throughout your career and in the real-world!). Feel free to use your favorite search engine to find more information about concepts we cover in this course — but be sure your work is your own! How else will you learn to code? Practice, practice, practice!