

Industrial Engineering & Operations Research

College of Engineering

IND ENG 210 Python for Analytics (3 units)

(subject to minor changes)

Course Description

Python programming language has become the de-facto standard in analytics applications. This introductory course provides students with sufficient background in Python programming language for use in analytics applications as well as potentially conducting research in the area. The course is designed to prepare students for the applied analytics problems and projects they will encounter in advanced analytics courses. It will start with basic programming topics using Python and cover using powerful Python packages such as Numpy, Scipy, Pandas, and Matplotlib that are essential for descriptive, predictive, and prescriptive analytics. Best practices of using modern AI tools to enhance programming productivity will also be discussed. Advanced data visualization tools will be covered through Seaborn and Plotly. Students will work on worksheets in groups in active learning sessions along with the instructor and also complete and present a final project in order to solidify the lectures into practical experience using Python for analytics.

Zoom link: <https://berkeley.zoom.us/j/9380095471>

Classroom: Blum Hall East, 3rd floor (3A)

LEARNING GOALS

Upon completion of the course students will have learned how to:

- Use Python and core scientific packages to solve complex analytics problems;
- Develop custom Python scripts and functions to manipulate data;
- Gain advanced data visualization skills for showcasing analytics results;
- Understand the array of mathematical toolkits provided by the Python packages covered.

COURSE PREREQUISITES

There are no prerequisites for this course but some prior programming experience may be

helpful.

WAIVER INFO

For those who are confident about their programming, Python, and data visualization skills, you may choose to take an exam (administered on **August 11**, online) to be considered as a waiver for the requirement of this course. The exam will broadly cover the content of the class – see the topics in the course outline section below as the exam topics. Receiving a score of 80% or higher on the exam will allow you to waive this course requirement.

TEXTBOOK(S) AND/OR OTHER REQUIRED MATERIAL

Python for Data Analysis, 3rd Edition by Wes McKinney, O'Reilly Media, Inc. Available online here: [Python for Data Analysis, 3E](#) In-class access to a computer capable of running Python is required.

COURSE OUTLINE

The course will run from August 18 to August 26. Days will be structured as having a lecture component and an active learning / group work component supervised and facilitated by the instructor. Lectures will begin 9:00am PT daily (45 mins lunch breaks will be provided as well). Below is an outline of the course schedule:

Monday, 9:00-15:00 (approx 3h lecture + 2.5h active learning / work time)

- Using Python: pros and cons
- Data Analytics vs. Data Analysis
- Discussing different IDEs and setup
- Installing the environment with core packages
- Working with the shell, IPython, and the editor
- Language specifics:
 - Basic arithmetic operations, assignment operators,
 - Data types: string, int, float, bool, list, dict
 - Control flow (if/elif/else)
 - Conditional expressions
 - Iterative programming (for/continue/while/break)
- Worksheet & group work time with the instructor
- Turn in Worksheets code
- Reading: Chapter 1

Tuesday, 9:00-15:00 (approx 3h lecture + 2.5h active learning / work time)

- Review of Monday
- Functions: definition, return values, pass by reference / mutable vs immutable objects
- Assertions
- Modules
- Classes
- Worksheet and group work time with the instructor
- Turn in Worksheets code

- Assignment 1 due on bCourses; 11:59pm

- Reading: Chapters 2-3.5

Wednesday, 9:00-15:00 (approx 3h lecture + 2.5h active learning / work time)

- Numpy intro
- Arrays: Indexing, slicing, reshaping
- Masking
- Vectorization
- Basic plotting with Matplotlib
- I/O and interaction with files
- Examples
- Worksheet and group work time with the instructor
- Turn in Worksheets code

- Assignment 2 due on bCourses; 11:59pm

- Reading: Rest of Chapter 3. Chapter 4

Thursday, 9:00-15:00 (approx 3h lecture + 2.5h active learning / work time)

- Review
- Pandas intro
- Dataframes
- File i/o with Pandas
- Compatibility with Numpy and other modules
- Jupyter Notebooks
- Examples & practice with datasets
- Worksheet and group work time with the instructor
- Turn in Worksheets code

- Assignment 3 due on bCourses; 11:59pm

Friday, 9:00-15:00 (approx 3h lecture + 2.5h active learning / work time)

- Useful Python Modules
- Scipy – stats and optimize
- Sklearn – machine learning in Python
- Worksheet and group work time with the instructor
- Project work time
- Turn in Worksheets code

- Assignment 4 due on bCourses; 11:59pm

Saturday, 9:00-10:30 (approx 1.5h lecture; + **Midterm Exam**)

- Review session
- Midterm Exam (you can take the midterm exam any time on Saturday, online)

Sunday 9:00-12:30 (3.5h lecture/mixed content)

- Coding with AI – useful tools and best practices; good prompting; working with multiple LLMs
- Better plotting with Plotly
- Finer details, useful tricks across Python
- Assignment 5 due on bCourses; 11:59pm

Monday 9:00-15:00 (3h lecture + 2.5h active learning / work time)

- Interactive Dashboards with Dash + Plotly
- Communication and presentation of data
- Worksheet and group work time with the instructor
- Using Git: version control and collaboration
- Turn in Worksheets code
- Project group work time

Tuesday 9:00-12:30 (3.5h - lecture, work time, presentations)

- Overview of other useful modules
- Integrations with other languages: Juliacall example
- Final project presentations

Grading

Grading will be based on homework assignments, a midterm, as well as a final project

- Assignments & Worksheets: 30%

- Midterm exam: 30%
- Final Project: 40%

Late assignments will not be accepted, since solutions will be discussed the following day. Instead, students will have one assignment dropped. Assignments are graded based on effort, not correctness, to disincentivize students from using tools such as ChatGPT in the first portion of the course. Exams are to be completed on time. Late final projects may incur a penalty of up to 50% depending on the delivery returned late.

Project Details:

Students will work in teams of 5 with a real dataset of their choice to solidify everything they have learned in the course. This will include data cleaning, exploratory data analysis, implementing best software development practices (modular programming, proper documentation, version control through Git), building a dashboard (via Plotly), and giving a short presentation about their work and showcasing their dashboard. While the lectures and homework assignments in the course will focus primarily on technical details and individual pieces of working with data in Python, the project will be about bringing these tools together in a cohesive and collaborative way to conclude the course.

Use of AI tools:

AI tools are playing an increasingly important role in the everyday work of many software developers, and are quickly becoming an integral part of the coding process. We will learn how to use these effectively in this course. However, for learning the fundamentals of Python, no use of AI tools will be allowed; you may, however, access any other online materials (code documentation, stackoverflow, textbooks, and so on) to help you with the worksheets and assignments (other than the midterm). From Sunday August 24 onwards, when we will have established our Python foundations and discuss coding with AI assistance, you may use any and all AI tools.

ACCOMMODATIONS

The purpose of academic accommodations is to ensure that all students have a fair chance at academic success. Disability, or hardships such as basic needs insecurity, uncertain documentation and immigration status, medical and mental health concerns, pregnancy and parenting, significant familial distress, and experiencing sexual violence or harassment, can affect a student's ability to satisfy particular course requirements. Students have the right to reasonable academic accommodations, without having to disclose personal information to instructors.

The [Academic Accommodations Hub](#) is the campus-wide portal for support resources and academic policies, including key accommodation information and links to mental health resources, resources for survivors of sexual violence and sexual harassment, and more. For College of Engineering specific information see [Engineering Student Resources](#).

[Disabled Students' Program \(DSP\)](#): If you have a disability, or think you may have a disability, you can work with DSP (260 César Chávez Student Center #4250; 510-642-0518;) to request an official accommodation. Services are individually designed and based on the specific needs of each student as identified by DSP's Specialists. If you have already been approved for accommodations through DSP, please contact the instructor to develop an implementation plan.

ONLINE DELIVERY

Mode of delivery The online form of teaching has proved its advantages in recent years so this course will deliver the material through well-established learning paths, providing the comfort of privacy for each individual student. The main instructional content will be delivered in live online webcast sessions with the students, where breakout room assignments will be given to groups or individually. Content will be supported by examples, and lecture slides that will be provided to the students. Live coding demos will demonstrate functionalities as they are learned. Asynchronous material will be provided for further reading. Solutions to specific problems discussed in class will be completed outside of class time. Additionally, students will be able to communicate with the instructor in face-to-face online chats with the instructor whenever needed. For those students interested in improving further, advanced material will be offered.

ADDITIONAL INFORMATION:

Relation to other IEOR courses IEOR's *Analytics Lab* course has a prerequisite for fluency in Python programming language. Students may demonstrate fluency by completing *Python for Analytics*. In the increased computer involvement in every scientific and professional field it is highly likely that this course can be used as a prerequisite to other courses as well. For example, the natural extension for this course would be to continue with studies of deeper

analytical tools while acquiring skills related to machine learning to create prototypes in the discovery of process and data dependencies.

Pedagogical reasoning The material of this course is necessary for analytics professionals who aim to possess programming skills to explore data through deep computer analysis and for prototype creation by using complex mathematical reasoning. Python is a versatile platform providing access to a plethora of scientific packages and easy manipulation of data.

Monitoring student progress Considering the delivery methods of the material, student learning progress will be assessed regularly through class participation, homework, breakout room (in online format only) individual and group assignments, problem solving, and delivery of the final project.

Evaluating learning outcomes Learning outcomes will be evaluated for each student through methods such as: in-class discussions of their work progress, revisions of specific material related to their work, final versions of written work and oral live in-class presentations in front of the class audience.

Additional consultations Course instructor(s) will develop, produce, and deliver instructional content through bCourses. Instructor(s) will foster learning by moderating discussion forums, providing feedback on students' work, and meeting students in person to comment on their work. They will be available for consultation both online and in-person.

Additional support There are a number of staff responsible for student services, information technology, communications in the department who support instructors and students. The program's student services advisors and IT staff have extensive experience with bCourses and GSuite. Head Graduate Advisor, Communications Director, Graduate Student Services Officer have extensive experience with conducting webinars.