

ISOM3400: Business Applications Development in Python

Winter 2024

Instructor: Dr. JIA, Jia (justinjia@ust.hk)
Office: LSK 5045
Office Hours: By appointment

Class Schedule

Lecture

Date	Time	Venue
Tue./Thu./Fri.	3:00PM ~ 6:20PM	LSK 1007

Lab

Date	Time	Venue
Tue./Thu./Fri.	2:00PM ~ 2:50PM	LSKG021

Course Website: <https://canvas.ust.hk>. All course materials and announcements will be posted on this site. You are advised to check it regularly throughout the course.

COURSE GOALS

This course will provide students with skills and knowledge of Python programming and experience in designing and developing business applications.

LEARNING OUTCOMES

By the end of this course, students will be able to:

- (1) Acquire general programming knowledge with Python language
- (2) Able to design and develop business applications in Python
- (3) Conduct programming with team members effectively

COURSE DESCRIPTION

Python has recently become the most popular general-purpose programming language according to many polls among programmers. The scripting nature of Python allows fast development and easy maintenance of applications. More importantly, the unparalleled community support makes Python increasingly powerful.

In this course, students will learn Python programming language in the context of business applications development. Business applications involve both business requirements and user requirements. Therefore, developers and programmers who design and develop business applications for organizations are required to meet those requirements. We believe that the combination of Python programming skills and business applications development will provide high practical value to students majoring in Information Systems as well as other related fields.

TEACHING APPROACH

In general, the teaching approach of this course is based on the notion of sustained, deep learning by applying knowledge through programming, hands-on practices, and assignments.

Teaching & Learning Activities	Roles in the Course	Learning Outcomes addressed
Lecture	Explain key concepts to students using an active learning approach, in-class exercise, and after-class discussion of questions.	1, 2, 3
Laboratory	Apply concepts presented in lectures to hands-on exercises.	1, 2, 3
Assignment	It requires students to apply their knowledge and understanding in programming to solve business analytics problems.	2, 3, 4

EVALUATION

Components	Percentage of the grade
A. In-class Exercises	25%
B. Assignments (Generative AI tools are allowed)	35%
C. Final Exam	40%
TOTAL:	100%

A. In-class Exercises (25%)

There are about 5 in-class exercises throughout the semester. They will give you hands-on practice in Python programming in a setting where you can ask questions and collaborate with fellow students. Students' answers will be collected and graded. All scores will count towards the final grade. **There will be NO makeup in-class exercises for whatever reasons.**

B. Assignments (Individual and group) (35%)

There are **TWO** assignments. Students are expected to apply Python programming skills to solve practical business application problems.

Assignment 1 (individual) - (15%) This is an individual assignment. Each student needs to submit his/her program by the deadline. The detailed grading criteria will be stated clearly in the assignment document.

Assignment 2 (group) - (20%) This is a group assignment. A pre-assigned group is required to design and develop a business application. A video demonstration (no more than 5 minutes) is required as part of assignment submission. Details of the group project will be provided later in the semester.

Peer evaluation

Peer evaluation will be conducted for the group assignment. Students should make sure they make a fair contribution.

Late policy

Turn in your work early if there is any uncertainty about your ability to turn it in at the due time. Submissions up to 24 hours late will have their grade reduced by 25%; those up to 48 hours late will have their grade reduced by 50%. They will not be accepted for credit after two days.

Honor-code policy

The basic presumption is that the work you submit is your own. Every line of text and line of code that you submit must be written by you personally.

However, occasionally, it may be necessary to ask someone for help. You are permitted to do so, provided you meet the following two conditions:

1. You acknowledge any help received on the work you hand in. That is, you must include a comment in your homework submission that clearly states the name of the student, book, or online reference from which you received assistance.
2. You *understand* the work that you hand in, so that you could explain the reasoning behind the parts of the work on which you received assistance from others.

We shall not deduct credit for small amounts of acknowledged assistance. Even working as a team on one of several problems in a problem set may not hurt your grade, as long as all members of the group acknowledge their collaboration. Such shared interest can be beneficial to all concerned. Nevertheless, we *do reserve the right* to give less than full credit in circumstances where it appears that there has been *large-scale division of labor*, and you are not getting as much learning out of the in-class exercise or assignment as you should.

Submissions that fail to properly acknowledge help from other students or non-class sources **will receive no credit**. Copied work **will receive no credit**. Any and all violations **will be reported** to the University administration.

Moreover, all students are expected to comply with the HKUST policy on academic integrity. This policy can be found online at <http://ugadmin.ust.hk/integrity/student-1.html>.

If you have any questions about what this policy means, please discuss the matter with the instructor.

C. Final Exam (40%)

There is a final exam (open note), which covers ALL topics taught in the semester. Details of the exam will be provided later in the semester.

D. Labs

In addition to in-class exercises, this course also has a complementary lab component, which further exposes you to more programming exercises. Although lab participation will not count towards the final grade, you are highly encouraged to make good use of lab hours to solve your puzzles and hone your programming skills.

MATERIALS

1. MAIN READING

This course has no required textbook. Lecture notes and extra exercises (either in pdf or ipynb format) will be posted on the course website.

2. SOFTWARE

- Anaconda Navigator (for Win-64, OSX-64, or Linux-64):
 - Jupyter notebook
 - Python 3.8+
- Google Colaboratory

OTHERS

Use of generative AI tools

Students are permitted to use Generative AI Tools such as ChatGPT for specific assignments, as designated by the instructor. Nonetheless, students are obligated to duly acknowledge and credit any use of generative AI. Note:

- Leveraging ChatGPT, individuals can effortlessly generate content devoid of grammatical errors. As a result, during the grading process, we will presume that the content is free of any grammatical blunders.
- Generative AI Tools can provide valuable assistance when seeking additional practice and examples. Students are encouraged to utilize them for self-study and honing programming skills.

Email Policy

Please put **[ISOM3400]** at the beginning of the subject line of your email along with your email **subject**. Failure to do so may result in a longer response time.

As expected, there will be numerous emails when it is closer to the due dates. If you need any assistance, raise them **as early as possible**, and/or take advantage of the office hours of the instructor and the TA. Note that **neither the instructor nor the TAs will provide direct answers to the assignments**.

Learning Environment

We welcome feedbacks on our teaching throughout the semester. You are encouraged to contact me or the TA any time you have any questions, suggestions, concerns, or would like to ask for advice.

TENTATIVE LECTURE SCHEDULE

DATE	TOPICS	ASSIGNMENTS/ DUE DATES
Jan. 4	Course Introduction Python Basics: Data, Data types, and Operators	
Jan. 5	Data Structures: Lists and Tuples	
Jan. 9	Data Structures: Dictionaries, and Sets In-class exercise session 1	
Jan. 11	Control Structures: if, for, while, and try statements	
Jan. 12	In-class exercise session 2 Functions and Classes I	Asg. 1 Release
Jan. 16	Functions and Classes II	
Jan. 18	In-class exercise session 3 Web Scraping with <i>Beautiful Soup</i>	Asg. 1 Due
Jan. 19	Web Automation with <i>Selenium</i>	Asg. 2 Release
Jan. 23	In-class exercise session 4 Web App Development I	
Jan. 25	Web App Development II	
Jan. 26	In-class exercise session 5 30-min Break Final Exam (4:50pm~6:50pm)	