

# ISOM3400: Business Applications Development in Python Spring 2025

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

## Teaching Assistant:

Mr. LEE, Aaron ([imaaron@ust.hk](mailto:imaaron@ust.hk))      Tel: 2358-7638      Office: LSK 4065  
Ms. WONG, Cetus ([imhtwong@ust.hk](mailto:imhtwong@ust.hk))      Tel: 2358-5728      Office: LSK 4048

## Class Schedule

### Lecture

Section	Date	Time	Venue
L1	Mon./Wed.	9:00AM - 10:20AM	LSK G012

### Lab

Section	Date	Time	Venue
LA1	Tu	4:30pm - 5:20pm	LSK G021
LA2	Tu	10:30am - 11:20am	
LA3	Mo	10:30am - 11:20am	
LA4	Mo	4:30pm - 5:20pm	

**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
A. Labs	5%
B. Assignment 1	15%
C. Assignment 2	15%
D. Final Exam – Programming concepts	25%
E. Final Exam – Advanced techniques	20%
F. Final Exam – Business applications	20%
<b>TOTAL:</b>	<b>100%</b>

### A. Labs

Participating in labs is a crucial part of your learning journey, as it provides hands-on experience with more programming exercises. You are expected to attend lab sessions to solve your puzzles, deepening your understanding, and hone your programming skills.

## ***B. Assignments***

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

### ***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 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.

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***

There is an open-book paper-based final exam, which covers ALL topics taught in the semester. It is further divided into 3 parts as listed in the evaluation section. Details of the exam will be provided later in the semester.

### ***Make-up policy***

There will be no make-up exams except due to extraordinary circumstances beyond your control such as medical emergencies. Students must submit required documentation described on the ARO's [Exam Regulations](#) website in order to be considered for a make-up exam.

## ***Grade appeal***

All scores will be uploaded to Canvas when ready. It is always the student's responsibility to check the scores and make sure they are correct. Any appeal to score has to be filed through email to [imaaron@ust.hk](mailto:imaaron@ust.hk) or [imhtwong@ust.hk](mailto:imhtwong@ust.hk). No appeal to a particular score is allowed 72 hours after its release.

## ***D. In-class Exercises***

There are 5 in-class exercise sessions scheduled throughout the semester. Most of the exercise questions are borrowed from past exams. They will help you reinforce programming concepts in a timely manner and gain hands-on programming experience in a setting where you can ask questions and collaborate with fellow students. **No answers from these sessions will be collected for grading.**

## **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 3.10+
- Visual Studio Code (VS code)
- 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 L3 LAX]** (X being the section number) **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 make good use of the labs and the office hours. Note that **neither the instructor nor the TAs will provide direct answers to the assignments.**

### *Learning Environment*

We welcome feedback 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
Feb 3	Course Introduction	
Feb 5	Python Basics: Data, Data types, and Operators	
Feb 10		
Feb 12	Data Structures: Lists, Tuples, Dictionaries, and Sets	Add/Drop deadline: Feb 15
Feb 17		
Feb 19	<b>In-class exercise session 1</b>	
Feb 24	Control Structures: if, for, while, and try statements	
Feb 26		
Mar 3		
Mar 5	<b>In-class exercise session 2</b>	Asg. 1 Release
Mar 10	Functions, Classes, and Modules	
Mar 12		
Mar 17		
Mar 19	<b>In-class exercise session 3</b>	
Mar 24	Web Scraping with <i>Beautiful Soup</i>	Asg. 1 Due
Mar 26		
Mar 31		
Apr 2	<i>Midterm Break</i>	
Apr 7	<b>In-class exercise session 4</b>	Asg. 2 Release
Apr 9	Data Analysis and Manipulation with <i>pandas</i>	
Apr 14	Web App Development with <i>Streamlit</i>	
Apr 16		
Apr 21	<i>Public Holiday</i>	
Apr 23		
Apr 28		Asg. 2 Due
Apr 30	<b>In-class exercise session 5</b>	
May 5	<i>Public Holiday</i>	
May 7	Final Revision	

### TENTATIVE LAB SCHEDULE

WEEK/DATE	TOPICS
Feb 3/4	Use of Colab, Setup of VS Code & Anaconda VSCode & Anaconda: Download, install and setup
Feb 10/11	Python Basics
Feb 17/18	Data Structures: List, Tuple
Feb 24/25	Data Structures: Dictionary, Set
Mar 3/4	Control Structures I
Mar 10/11	Control Structures II
Mar 17/18	Function and Class I
Mar 24/25	Web Scraping I: HTML and CSS Basics
Mar 31/Apr 1	<i>Midterm Break</i>
Apr 7/8	Web Scraping II: Beautiful Soup

*\* Note both the lecture, lab, and assignment schedules are tentative and subject to change without notice.*