ISOM 3400 - Business Applications Development in Python

Spring 2026

Instructor: Jing WANG, Ph.D.
Email: jwang@ust.hk Begin subject: [ISOM3400]
Tel: 3469-2125
Office: LSK 4044
Office Hours: By appointment

Teaching Assistant: William YAP
Email: imwilliam@ust.hk Begin subject: [ISOM3400]
Tel: 2358-7656
Office: LSK 4048
Office Hours: By appointment

Class Schedule

Lecture

Section	Date	Time	Venue
L2	Tu	4:30pm – 5:50pm	G009A, CYT Bldg
	Th		
L3	Tu	3:00pm-4:20pm	Rm 1011, LSK Bldg
	Th		

Lab

Section	Date	Time	Venue
LA2	Mo	9:00am – 9:50am	LSK G021
LA3	Mo	10:30am – 11:20am	LSK G021
LA4	Tu	9:00am – 9:50am	LSK G021

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 DESCRIPTION

Python has recently emerged as the most popular general-purpose programming language, according to many polls among programmers. Its scripting nature allows fast development and

easy maintenance of applications. More importantly, the unparalleled community support makes it increasingly powerful.

In this course, students will learn the Python programming language in the context of business application development. Business applications must address both organizational objectives and user requirements; therefore, developers are expected to design and implement solutions that effectively meet these needs. We believe that combining Python programming skills with business application development will provide high practical value for students majoring in Information Systems as well as those in related fields.

INTENDED LEARNING OUTCOMES

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

(1) Attain Proficiency in Python Programming - Develop a solid understanding of programming concepts using the Python language.
(2) Capability to Design and Develop Python-based Business Applications - Acquire the skills necessary to design and construct business applications utilizing Python.
(3) Effective Collaborative Programming - Cultivate the ability to collaborate efficiently with team members while engaging in programming tasks.

EVALUATION

Components	Percentage
A. In-class Activities	10%
B. Assignment	20%
C. Final Exam – Programming logic	20%
D. Final Exam – Advanced techniques	25%
E. Final Exam – Business applications	25%
TOTAL:	**100%**

A. In-class Activities

Your performance for in-class activities is based on 5 in-class exercise submissions and 2 mandatory lab participation (please refer to the tentative schedule).

The five in-class exercise sessions are scheduled throughout the semester and mainly feature questions adapted from past exams. These sessions are designed to reinforce programming concepts promptly while giving you hands-on experience in an interactive environment where you can ask questions and collaborate with your peers.

Attendance and active participation in the lab sessions are equally important. Labs offer additional hands-on programming exercises that allow you to deepen your understanding, solve challenging problems, and further develop your programming skills.

B. Assignments

There are **ONE** individual assignment released during the semester. You are expected to apply your Python programming skills to solve real-world business challenges.

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 or use generative AI tools. 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.

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.

C. Final Exam

There is an **open-note paper-based** final exam, which covers ALL topics taught in the semester. It is further divided into 3 parts.

- **Programming Logic:** This part will focus on evaluating your understanding of programming logic.
- **Advanced Techniques**: This part will test you on advanced programming techniques, drawing from the full range of topics discussed during the semester.
- **Business Applications**: This part is dedicated to business applications, assessing your ability to apply programming concepts in practical business contexts.

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 appropriate documentation issued by a registered medical practitioner 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 must be filed through email to jwang@ust.hk. No appeal to a particular score is allowed 72 hours after its release.

MATERIALS

1. MAIN READING

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

2. SOFTWARE

- Google Collaboratory
- Anaconda Navigator
    - Jupyter notebook
    - Python 3.11+
- Visual Studio Code (VS code)

Feedback

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

TENTATIVE LECTURE SCHEDULE

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

TENTATIVE LAB SCHEDULE

WEEK/DATE	TOPICS
Feb 2/3	Use of Colab, Setup of VS Code & Anaconda VSCode & Anaconda: Download, install, and setup
Feb 9/10	Python Basics
Feb 16/17	*Public Holiday*
Feb 23/24	Data Structures: List, Tuple
Mar 2/3	Data Structures: Dictionary, Set
Mar 9/10	Control Structures I
Mar 16/17	Control Structures II
Mar 23/24	Function and Class I
Mar 30/31	Web Scraping I: HTML and CSS Basics (mandatory)
Apr 6/7	*Midterm Break*
Apr 13/14	Web Scraping II: Beautiful Soup

Apr 20/21	Streamlit Setup (mandatory)
Apr 27/28	Web App Development I
May 4/5	Web App Development II

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