

ISOM3400: Business Applications Development in Python**3 Credits****Prof. James Kwok**jkwok@ust.hk**Office Hours: By appointment****Course Goals**

This course provides students with foundational skills in Python programming and hands-on experience in designing and developing business applications.

Intended Learning Outcomes (ILOs)

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

1. Understand fundamental programming concepts in Python.
2. Apply Python syntax and control structures to solve business problems.
3. Build business applications using Python frameworks.

Course Description

Python has become the leading general-purpose programming language, known for its simplicity, rapid development capabilities, and strong community support. This course introduces Python in the context of business application development, emphasizing the importance of aligning technical solutions with business and user requirements.

Students majoring in Information Systems and related fields will gain practical value from combining Python expertise with business application development skills. As a programming-focused course, students are expected to independently explore online resources and research Python syntax beyond the provided materials to complete assignments and projects.

Assessment and Grading

This course will be assessed using **criterion-referencing** and grades will not be assigned using a curve. Detailed rubrics for each assignment are provided below, outlining the criteria used for evaluation.

Summary Table

Assessment Task	Contribution to Overall Course Grade (%)	Due Date
A. Lab Attendance	5%	As per course schedule
B. In-Class Exercise	10%	As per course schedule
C. Assignment	15%	As per course schedule
D. Midterm Exam	20%	As per course schedule
E. Final Exam — Advanced Techniques	25%	To be announced
F. Final Exam — Business Applications	25%	To be announced

A. Lab Attendance (5%)

Students will receive full marks for lab attendance by attending **any six** lab sessions, representing approximately 50%–60% of the scheduled lab sessions. Students who attend fewer than six lab sessions will receive a mark of zero for lab attendance.

B. In-Class Exercise (10%)

Students will complete **FIVE** individual in-class exercises during the semester. Please note that **no make-up opportunities** will be provided under any circumstances.

Each student is expected to apply their Python programming skills independently to solve a business-related problem using Visual Studio Code (VS Code) and Google Colab. At the end of the session, individual submissions will be collected and assessed. Students must submit their own work and demonstrate meaningful individual effort to receive credit.

Technical Requirements

Students are responsible for ensuring that VS Code, Google Colab, and all required Python packages are properly installed and functioning prior to the exercise.

Late Submission Policy:

Late submissions for the in-class exercise within **10 minutes** will result in a **30% mark** deduction. Submissions **more than 10 minutes late will not be accepted**.

C. Assignment (15%)

This course includes an individual assignment requiring students to apply their Python programming skills to solve real-world business challenges. There is only one individual assignment. The specifics of the assignment will be communicated at a subsequent point in the course.

Selected students will showcase their findings during the final class session through the presentation of their video files if possible.

Late Submission Policy:

Late submissions for the individual assignment within **3 hours** will result in a **30% mark** deduction. Submissions **more than 3 hours late will not be accepted**.

D. Midterm Exam (20%)

The midterm exam will cover all topics taught up to that point. Please note that **no make-up opportunities** will be provided under any circumstances.

E. Final Exam — Advanced Techniques (25%)

This cumulative section of the final exam covers all lecture and lab materials taught during the semester and assesses students' ability to apply advanced programming techniques across the course topics.

F. Final Exam — Business Applications (25%)

This cumulative section of the final exam covers all lecture and lab materials taught during the semester and assesses students' ability to apply programming concepts in practical business contexts and develop solutions aligned with business needs.

Arrangements for the Make-up Final Exam

Make-up final exams will only be accommodated in exceptional circumstances beyond a student's control (e.g., medical emergencies). Students requesting a make-up exam must submit official supporting documentation, specifically the university's request form (**Form EX-16**) and a medical certificate, to the course instructor via email. The make-up exam will be in essay format.

(Attention: Students who are eligible to take the make-up exam are required to write a research article that includes an introduction, references, proper citations, and other essential sections. The article must be completed within a few hours of being assigned. Students are expected to apply the knowledge and skills acquired in this course when completing the make-up exam. Please note that there will be **no opportunity for a second make-up exam** under any circumstances. Failure to submit the research article for any reason, including email or internet issues, will result in a grade of ZERO for the exam.)

Final Exam Paper Checking

After the Final Examination papers have been graded, the corresponding scores and detailed marking records will be uploaded to Canvas. Students are required to review and verify their scores promptly. To comply with the University's final grade submission deadlines, **a two-hour paper-checking session** will be arranged and facilitated by the Teaching Assistant. During this session, students may inspect the marking of their examination papers in person. Once the paper-checking session has concluded, all assessment scores will be considered final, and no further modifications or adjustments will be permitted under any circumstances.

Feedback on all assessments:

- Feedback on all assignments and assessments will be provided within **10 working days**.
- A summary highlighting common mistakes or key deficiencies in answering questions will be shared with students.
- Additionally, students can schedule a meeting with our Teaching Assistant (TA) to review their assignments and examination papers, gaining insights into their mistakes and deficiencies. This review session must take place within a specified deadline, typically two working days after the scores are released. After this deadline, students **will not be allowed** to review their assignment and examination papers.

Score Posting and Appeal Policy

Upon completion, all scores will be posted on Canvas. It is incumbent upon the student to review their scores and verify their accuracy. If any discrepancies arise, score appeals must be submitted via email to jkwok@ust.hk. It is important to note that score appeals will not be entertained once the designated checking/appeal period has elapsed (e.g., two working days after the score release) if applicable.

[If a student is unable to review their paper within the designated checking period, the score will be considered final by default. Once the checking or appeal period has ended, all scores will be finalized, and no further modifications or corrections will be permitted.]

Mapping of Course ILOs to Assessment Tasks

Assessment Task	Mapped ILOs	Explanation
A. Lab Attendance	ILO1, ILO2	This task requires students to attend laboratory sessions and engage with the lab materials to support their learning of Python programming and business application development.
B. In-Class Exercise	ILO1, ILO2	This task assesses students' understanding of fundamental programming concepts in Python (ILO1) and their ability to apply Python syntax and control structures to solve business problems independently (ILO2). Students are expected to complete individual coding tasks during class and submit their own work for assessment.
C. Assignment	ILO1, ILO2, ILO3	This task assesses students' understanding of fundamental programming concepts in Python (ILO1), their ability to apply Python syntax and control structures to solve business problems (ILO2), and their ability to build business applications using Python frameworks (ILO3).
D. Midterm Exam	ILO1, ILO2	This section evaluates students' understanding of fundamental Python programming concepts (ILO1) and their ability to apply Python syntax and control structures to solve business problems (ILO2).
E. Final Exam — Advanced Techniques	ILO1, ILO2	This section assesses students' ability to apply Python syntax, control structures, and advanced programming techniques to solve business problems (ILO2), while also drawing on fundamental Python concepts (ILO1).
F. Final Exam — Business Applications	ILO2, ILO3	This section assesses students' ability to apply Python syntax and control structures to solve business problems (ILO2) and build business applications using Python frameworks (ILO3). It emphasizes practical business application development and alignment with business needs.

Final Grade Descriptors:

Grades	Short Description	Elaboration on subject grading description
A	Excellent Performance	Demonstrates exceptional understanding of fundamental programming concepts in Python, consistently applies Python syntax and control structures to solve complex business problems, and builds effective business applications using Python frameworks. Shows strong programming logic, practical problem-solving ability, and well-structured application design.
B	Good Performance	Shows solid understanding of fundamental programming concepts in Python and applies Python syntax and control structures effectively to solve business problems. Demonstrates good ability to build functional business applications using Python frameworks.
C	Satisfactory Performance	Possesses basic understanding of fundamental programming concepts in Python and can apply Python syntax and control structures to solve standard business problems. Demonstrates adequate ability to build simple business applications using Python frameworks, though may require assistance.
D	Marginal Pass	Has limited understanding of fundamental programming concepts in Python and struggles to apply Python syntax and control structures to solve business problems. Demonstrates weak ability to build business applications using Python frameworks.
F	Fail	Demonstrates insufficient understanding of fundamental programming concepts in Python and fails to apply Python syntax and control structures to solve business problems. Lacks the ability to build usable business applications using Python frameworks.

Student Rubrics

Use the following rubrics to guide you for the assessment tasks that you submit in this course.

Assessment: Lab Attendance

Criteria: Lab Attendance
• Full Marks: Attends at least six lab sessions and engages with the corresponding lab materials. • Zero Marks: Attends fewer than six lab sessions.

Assessment: In-Class Exercise

Criteria: Application of Python Concepts
• Excellent: Demonstrates thorough understanding and applies Python concepts accurately and creatively. • Good: Applies Python concepts correctly with minor errors or omissions. • Satisfactory: Applies basic Python concepts with noticeable errors or lack of depth. • Marginal: Struggles to apply Python concepts effectively. • Fail: Fails to apply Python concepts or demonstrates misunderstanding.
Criteria: Individual Participation and Effort
• Excellent: Actively works on the exercise throughout the session, demonstrates strong individual effort, and submits a complete and thoughtful solution. • Good: Works consistently on the exercise and submits a mostly complete solution with clear individual effort. • Satisfactory: Makes a reasonable attempt but the submission is basic or incomplete in some areas. • Marginal: Shows limited individual effort and submits work with substantial gaps. • Fail: Does not participate meaningfully or does not submit an individual solution.
Criteria: Meeting Stated Requirements
• Excellent: Fully meets all requirements with a robust and complete solution. • Good: Meets most requirements with minor omissions. • Satisfactory: Meets basic requirements but lacks completeness. • Marginal: Partially meets requirements; key elements are missing. • Fail: Fails to meet the majority of requirements.

Assessment: Assignment

Criteria: Use of Python
• Excellent: Advanced use of Python with efficient, well-structured code and appropriate libraries. • Good: Correct use of Python with mostly efficient and readable code. • Satisfactory: Basic use of Python with some inefficiencies or structural issues. • Marginal: Limited use of Python with significant issues in code structure or logic. • Fail: Incorrect or missing use of Python in the assignment.
Criteria: User Friendliness
• Excellent: Highly intuitive and user-friendly interface with clear instructions and smooth navigation. • Good: Mostly user-friendly with minor usability issues. • Satisfactory: Basic usability with some confusing elements or unclear instructions. • Marginal: Poor usability with significant navigation or instruction issues. • Fail: Unusable interface or lack of user guidance.
Criteria: Meeting Stated Requirements
• Excellent: Fully meets and exceeds all stated requirements with innovative enhancements. • Good: Meets all stated requirements with minor omissions. • Satisfactory: Meets most requirements but lacks completeness or clarity. • Marginal: Meets few requirements with major omissions. • Fail: Does not meet the stated requirements.
Criteria: Presentation
• Excellent: (a) Presentation is clear, engaging, and professional with excellent video and audio quality. (b) Demonstrates thorough preparation and understanding of the material. • Good: (a) Presentation is clear but may have slight issues with video or audio quality. (b) Covers most key points effectively, but minor details could be enhanced. • Satisfactory: (a) Presentation is understandable but lacks engagement or quality in video/audio. (b) Some key points are covered, but it feels incomplete or rushed. • Marginal: (a) Presentation is unclear or poorly organized, with major issues in video or audio quality. (b) Fails to engage the audience or misses significant content. • Fail: (a) Presentation is unclear or unprofessional, with very poor video/audio quality. (b) Lacks adequate coverage of the topic, showing little effort or preparation.

Assessment: Midterm Exam

Criteria: Detecting and Correcting Errors
• Excellent: Accurately detects and corrects all errors in the code, demonstrating a clear understanding of program logic and effective troubleshooting. • Good: Detects and corrects most errors but misses a few minor issues or overlooks some details. • Satisfactory: Identifies and corrects some errors, though several may remain, affecting the overall functionality of the program. • Marginal: Detects and corrects only a few errors, with many issues still affecting the program's execution. • Fail: Fails to detect or correct most errors, resulting in a program that does not function as intended.
Criteria: Predicting Outputs
• Excellent: Accurately predicts the outputs for all test cases, providing detailed reasoning behind predictions and demonstrating deep understanding of program flow. • Good: Predicts the outputs for most test cases, with minor errors in reasoning or predictions. • Satisfactory: Provides reasonable output predictions but may misinterpret some program behavior or overlook edge cases. • Marginal: Predicts outputs inaccurately, with significant errors in reasoning or a misunderstanding of program logic. • Fail: Fails to predict outputs correctly, with no understanding of the underlying program flow.
Criteria: Revising and/or Enhancing Code
• Excellent: Makes insightful revisions and enhancements to improve code efficiency, readability, and functionality, adhering to best practices in programming. • Good: Makes adequate revisions and enhancements, but some areas of the code could be further optimized or improved. • Satisfactory: Makes basic revisions or enhancements, but the code still lacks efficiency or clarity in some parts. • Marginal: Makes minimal revisions or enhancements, with no noticeable improvement in code quality. • Fail: Fails to make any meaningful revisions or enhancements to the code, leaving it inefficient or non-functional.

Assessment: Final Exam — Advanced Techniques

Criteria: Application of Advanced Programming Techniques
• Excellent: Applies advanced techniques such as OOP effectively. • Good: Uses advanced techniques with minor implementation issues. • Satisfactory: Basic use of advanced techniques with limited understanding. • Marginal: Attempts advanced techniques with poor execution. • Fail: Fails to apply advanced programming techniques.
Criteria: Effective Use of Programming Practices
• Excellent: Demonstrates best practices including modularity, documentation, and testing. • Good: Applies good practices with minor omissions. • Satisfactory: Applies basic practices with noticeable gaps. • Marginal: Limited use of programming practices. • Fail: Poor or no use of programming practices.
Criteria: Meeting Stated Requirements
• Excellent: Fully meets and enhances all requirements with advanced features. • Good: Meets all requirements with minor gaps. • Satisfactory: Meets most requirements with basic implementation. • Marginal: Meets few requirements with major issues. • Fail: Does not meet the stated requirements.

Assessment: Final Exam — Business Applications

Criteria: Solution Development and User Requirements
• Excellent: Develops comprehensive solutions aligned with user needs and business context. • Good: Develops appropriate solutions with minor gaps in user alignment. • Satisfactory: Basic solution development with limited user consideration. • Marginal: Incomplete solutions with poor user alignment. • Fail: Fails to develop a usable solution.
Criteria: Application of Programming Techniques
• Excellent: Effectively applies programming techniques to solve business problems. • Good: Applies techniques with minor inefficiencies. • Satisfactory: Basic application with limited problem-solving effectiveness. • Marginal: Poor application with minimal relevance. • Fail: Fails to apply programming techniques appropriately.
Criteria: Effective Use of the Programming Language
• Excellent: Demonstrates mastery of Python with efficient and readable code. • Good: Uses Python effectively with minor issues. • Satisfactory: Basic use of Python with noticeable inefficiencies. • Marginal: Limited use of Python with major issues. • Fail: Fails to use Python effectively.
Criteria: Meeting Stated Requirements
• Excellent: Fully meets and exceeds all business application requirements. • Good: Meets all requirements with minor gaps. • Satisfactory: Meets most requirements with basic implementation. • Marginal: Meets few requirements with major issues. • Fail: Does not meet the stated requirements.

Efficient Email Communication Guidelines

To ensure prompt assistance, please include [**Course Code - LX**] (**X** being the section number), e.g., [**ISOM3400-L1**] at the start of your email's subject line. Neglecting this may lead to delays in our response time.

Anticipate a surge in email volume as deadlines approach. For timely support, address your queries ahead of time and utilize instructor and TA office hours.

Kindly note that **direct assignment answers won't be furnished by the instructor or TAs**. Your understanding and collaboration are appreciated.

Policy on the Use of Generative AI

Students are permitted to utilize generative artificial intelligence (AI) tools exclusively for enhancing programming tasks within this course. Nonetheless, students are obligated to duly acknowledge and credit any employment of generative AI. In the context of producing **video presentations, employing generative AI tools is strictly prohibited** for students.

Leveraging Gen AI tools, individuals can effortlessly generate content devoid of grammatical errors. As a result, during assessment, we presuppose that the content is devoid of any grammatical blunders. The following table summarizes where generative AI tools may be used in this course.

Course Activity	Use of Gen AI Tools
In-Class Exercise	Allowed
Assignment	Allowed
Midterm and Final Exam	Not allowed
Lecture and Lab	Allowed
Outside the Class for Learning	Allowed

We anticipate students to acquire coding skills by independently employing Gen AI tools. For instance, when seeking additional practice and examples, Gen AI tools can provide valuable assistance.

Student Learning Resources

Text and Reference Books

No particular textbooks or reference books are mandatory for this course. The learning materials will comprise diverse readings accessible on Canvas.

Course Website

Course content updates and other pertinent information will be communicated through the course website (URL to be announced) and Canvas (<http://canvas.ust.hk>). It is advisable for students to consistently monitor this platform throughout the semester.

Software Requirements

- Python 3.13+
- Google Colab
- Visual Studio Code (VS Code)
- Gen AI, e.g., Copilot

Course Schedule

This course is offered in lecture sessions and laboratory sessions.

Session	Day	Time	Venue
L1	Tuesday/Thursday	16:30 – 17:50	LSK1003
LA1A	Monday	12:30 – 13:20	LSK G021
LA1B	Wednesday	13:30 – 14:20	LSK G021

Schedule of Lecture (Tentative)

Week	Date	Topics	Assignment Release/Due
1	1 Sept	Introduction to Course and Programming Introduction to Python and Business Applications Gen AI	
	3 Sept	Python Basics: Variables, Data Types, Operators, String, Inputs and Outputs	
2	8 Sept	Data Structure: Lists, Tuples, and Dictionaries	
	10 Sept	Data Validation: If-else, for, while, try-except	Add/Drop Deadline: 14 Sept
3	15 Sept	Data Validation: If-else, for, while, try-except In-Class Exercise 1	
	17 Sept	Examples of Business Application	
4	22 Sept	Data Validation: If-else, for, while, try-except In-Class Exercise 2	
	24 Sept	Functions and Classes	
5	29 Sept	Functions and Classes	
	1 Oct	No Class: National Day	
6	6 Oct	Practice	
	8 Oct	Class Cancelled: Midterm Exam	Midterm Exam: (Date TBC)
7	13 Oct	Functions and Classes	
	15 Oct	Functions and Classes	
8	20 Oct	Functions and Classes In-Class Exercise 3	
	22 Oct	Web Automation — Beautiful Soup	
9	27 Oct	Web Automation — Beautiful Soup	Assignment: Release on 27 Oct
	29 Oct	Web Automation — Beautiful Soup	
10	3 Nov	Web Automation — Beautiful Soup In-Class Exercise 4	
	5 Nov	Web Applications Development — Streamlit	
11	10 Nov	Web Applications Development — Streamlit	
	12 Nov	Web Applications Development — Streamlit	
12	17 Nov	Web Applications Development — Streamlit	
	19 Nov	Web Applications Development — Streamlit In-Class Exercise 5	
13	24 Nov	Web Applications Development — Streamlit	Assignment: Due on 25 Nov, 17:00
	26 Nov	Revision	

Schedule of Laboratory (Tentative)

Week	Date	No.	Topics
1	7/9 Sept	LA01	VSCode & Anaconda, Gen AI, Virtual Environment, Data Types
2	14/16 Sept	LA02	String, Lists, Tuples, and Dictionaries
3	21/23 Sept	LA03	Data validation practice: if-else, for, while, try-except
4	28/30 Sept	LA04	Functions and Classes
5	5/7 Oct	LA05	Midterm Revision
6	12/14 Oct	LA06	Functions and Classes
7	19/21 Oct	LA07	No Lab on 19 Oct: Chung Yeung Festival Functions and Classes
8	26/28 Oct	LA08	Web Automation – Beautiful Soup
9	2/4 Nov	LA09	Web Automation – Beautiful Soup
10	9/11 Nov	LA10	Web Applications Development – Streamlit
11	16/18 Nov	LA11	Web Applications Development – Streamlit
12	23/25 Nov	LA12	Web Applications Development – Streamlit
13	30 Nov	LA13	Revision

***Note:** If lab sessions are canceled, a Zoom recording will be provided when available.

Contact Details for Instructor and TA

Prof. Kwok's office is located in room LSK4080, and he extends a warm invitation for you to visit during his office hours or at your convenience for any queries you may have. For urgent concerns, feel free to reach out via email (jkwok@ust.hk) or phone (2358-7652); however, he does emphasize that email is the preferred mode of communication as he frequently monitors it. Additionally, the Teaching Assistant (TA) assigned to this course is available to address inquiries related to grading, attendance, assignments, and any administrative matters.

Academic Honesty

Upholding academic integrity stands as a fundamental principle within our university community. Any breach of integrity undermines the foundation of our learning environment and the essence of inquiry that is vital for the institution's effectiveness. I maintain a zero-tolerance stance towards cheating, and no exceptions will be entertained. Students found engaging in acts of cheating, plagiarism, or any form of academic dishonesty will face a reduction of their course grade by a minimum of one letter grade. Moreover, it is my responsibility to report any instances of unethical conduct or indications of dishonesty in this course to the University.

Please bear in mind the current university regulation: any occurrence of cheating, irrespective of its magnitude, will result in an "X" grade notation on the student's academic record, signifying that the grade was attained through dishonest means. This "X" grade will persist on the student's record until graduation. Should a student be caught cheating again and subsequently receive another "X" grade, they will be dismissed from the University.

Plagiarism encompasses the act of copying text or ideas from external sources without appropriate citation. Even if you rephrase the concept using your own words, citing the origin is necessary when utilizing someone else's idea. It is imperative to exercise extreme caution to prevent presenting someone else's work as your own. Proper citations are obligatory when incorporating external sources' ideas, arguments, or any content. Whether drawing from research or the Internet, it is mandatory to acknowledge the source, even if you employ the general notion rather than verbatim wording.

Learning Environment

I wholeheartedly embrace feedback on my teaching during the entirety of the semester. I strongly encourage you to reach out to me or my TA whenever you have questions, suggestions, concerns, or if you seek advice. Your input is valued and will contribute to enhancing the learning experience. Feel free to contact us at your convenience.