

City University of Hong Kong
Course Syllabus

offered by Department of Information Systems
with effect from Semester A 2018 / 2019

Part I Course Overview

Course Title: Python Programming for Business

Course Code: IS2240

Course Duration: One Semester (13 weeks)

Credit Units: 3

Level: B2

Proposed Area:
(for GE courses only)

Arts and Humanities
 Study of Societies, Social and Business Organisations
 Science and Technology

Medium of Instruction: English

Medium of Assessment: English

Prerequisites:
(Course Code and Title) Nil

Precursors:
(Course Code and Title) Nil

Equivalent Courses:
(Course Code and Title) Nil

Exclusive Courses:
(Course Code and Title) Nil

Part II Course Details

1. Abstract

(A 150-word description about the course)

Python is a popular open source programming language widely used by business professionals to develop applications in big data, artificial intelligence, financial technology, and so on. It is easy to learn and fun to use. This course provides an in-depth introduction to the Python language and helps students master its fundamentals and apply them in different contexts to solve business problems. Students without programming background are also welcome. Students' skill set upon course completion should include implementing decision-making structure, repetition structure, function procedures, web scraping, database connection, testing, and debugging, among others.

2. Course Intended Learning Outcomes (CILOs)

(CILOs state what the student is expected to be able to do at the end of the course according to a given standard of performance.)

No.	CILOs [#]	Weighting* (if applicable)	Discovery-enriched curriculum related learning outcomes (please tick where appropriate)		
			A1	A2	A3
1.	Explain the structure of a Python program and understand the basics of computer programming.	20%	✓		
2.	Read, analyze, test and debug Python programs.	20%	✓	✓	
3.	Identify, characterize, and analyze a problem, and write Python programs to solve the business problem.	30%		✓	✓
4.	Apply Python programming knowledge and techniques to address a business problem, which involve advanced skills such as database connection, web scraping, and so on.	30%		✓	✓
		100%			

* If weighting is assigned to CILOs, they should add up to 100%.

[#] Please specify the alignment of CILOs to the Gateway Education Programme Intended Learning outcomes (PILOs) in Section A of Annex.

A1: Attitude

Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability

Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to self-life problems.

A3: Accomplishments

Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

3. Teaching and Learning Activities (TLAs)

(TLAs designed to facilitate students' achievement of the CILOs.)

TLA	Brief Description	CILO No.				Hours/week (if applicable)
		1	2	3	4	
TLA1: Lecture	Concepts and general knowledge of programming techniques in Python are explained.	✓	✓	✓	✓	Lecture: 1 Hour/Week
TLA2: Laboratory Exercise	Hands-on computer exercises related with business domains are designed to help students apply what they have learned in lecture. Major assignment involves individual work or teamwork by a group of students in same laboratory group to solve a specific business problem.		✓	✓	✓	Laboratory: 2 Hours/Week
TLA3: Tutorial	Concepts, techniques, and good practices of programming are discussed.	✓	✓	✓	✓	
TLA4: Class Discussion and Presentation	Perform online quizzes in lecture, tutorial, and laboratory to get immediate feedback from students. This is followed by discussion of the quizzes afterwards to reinforce the learning of the materials tested. Presentation of laboratory results and assignments may be required.	✓	✓	✓	✓	

4. Assessment Tasks/Activities (ATs)

(ATs are designed to assess how well the students achieve the CILOs.)

Assessment Tasks/Activities	CILO No.				Weighting*	Remarks [#]
	1	2	3	4		
Continuous Assessment: <u>60%</u>						
<u>AT1: Participation and Laboratory Exercises</u> Each laboratory has in-class exercises to assess students' hands-on programming skills of the topics covered.	✓	✓	✓	✓	20%	
<u>AT2: Individual Assignment</u> The individual assignment, including programme codes, results, written report and presentation, is required to assess the technical analysis and implementation skill sets of the students.		✓	✓	✓	20%	
<u>AT3: Mid-Term Test</u> The quizzes serve the purpose of continuous assessment of students' understanding of the key domain areas and as an indicator of how well the students have performed.	✓	✓	✓	✓	20%	
Examination: <u>40%</u> (duration: one 2-hour exam)						
<u>AT4. Final Examination</u> An examination to examine the students' ongoing learning and understanding of the subject matter.	✓	✓	✓	✓	40%	
* The weightings should add up to 100%.					100%	

[#] Remark: Students must pass BOTH coursework and examination in order to get an overall pass in this course.

5. Assessment Rubrics

(Grading of student achievements is based on student performance in assessment tasks/activities with the following rubrics.)

Assessment Task (AT)	Criterion	Excellent (A+, A, A-)	Good (B+, B, B-)	Fair (C+, C, C-)	Marginal (D)	Failure (F)
AT1: Participation and Laboratory Exercises	Ability to accurately describe and understand the basic concepts in Python programming	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to quickly understand and analyze a Python program	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to creatively, effectively and efficiently write Python programs	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Capability to creatively and effectively develop applications that involve advanced techniques to solve business problems	High	Significant	Moderate	Basic	Not even reaching marginal levels
AT2: Individual Assignment	Ability to effectively test and debug Python programs	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to creatively, effectively and efficiently write Python programs	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Capability to creatively and effectively develop applications that involve advanced techniques to solve business problems	High	Significant	Moderate	Basic	Not even reaching marginal levels
AT3: Mid-Term Test	Ability to accurately describe and understand the basic concepts in Python programming	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to accurately understand and analyse a Python program	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to creatively, effectively and efficiently write Python programs	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Capability to creatively and effectively develop applications that involve advanced techniques to solve business problems	High	Significant	Moderate	Basic	Not even reaching marginal levels
AT4: Final Examination	Ability to accurately describe and understand the basic concepts in Python programming	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to accurately understand and analyse a Python program	High	Significant	Moderate	Basic	Not even reaching marginal levels
	Ability to creatively, effectively and efficiently write Python programs	High	Significant	Moderate	Basic	Not even reaching marginal levels

	Capability to creatively and effectively develop applications that involve advanced techniques to solve business problems	High	Significant	Moderate	Basic	Not even reaching marginal levels
--	---	------	-------------	----------	-------	-----------------------------------

Part III Other Information

1. Keyword Syllabus

(An indication of the key topics of the course.)

Programming basics: variables, data types, lists, if statements, loops, functions, testing, debugging, and so on.

Advanced topics and business applications: web scraping, database connection, and so on.

2. Reading List

2.1 Compulsory Readings

(Compulsory readings can include books, book chapters, or journal/magazine articles. There are also collections of e-books, e-journals available from the CityU Library.)

1.	David I. Schneider, <u>An Introduction to Programming Using Python</u> , 1st edition, Pearson, 2016.
----	--

2.2 Additional Readings

(Additional references for students to learn to expand their knowledge about the subject.)

1.	Cay S. Horstmann, Rance D. Necaise, <u>Python for Everyone</u> , 2nd Edition, Wiley, 2016.
2.	Mark Lutz, <u>Learning Python</u> , 5 th Edition, O'Reilly Media, 2013.
3.	Eric Matthes, <u>Python Crash Course: A Hands-On, Project-Based Introduction to Programming</u> , 1 st Edition, No Starch Press, 2015.
4.	Al Sweigart, <u>Automate the Boring Stuff with Python: Practical Programming for Total Beginners</u> , 1 st Edition, No Starch Press, 2015.
5.	Mahesh Venkitachalam, <u>Python Playground: Geeky Projects for the Curious Programmer</u> , 1 st Edition. No Starch Press, 2015.