

This Version is No Longer Current
The latest version of this module is available [here](#)

MODULE DESCRIPTOR

Module Title

Fundamentals of Programming

Reference	CM1705	Version	3
Created	June 2022	SCQF Level	SCQF 7
Approved	May 2019	SCQF Points	30
Amended	July 2022	ECTS Points	15

Aims of Module

To provide students with an understanding of fundamental programming principles and concepts to solve business problems.

Learning Outcomes for Module

On completion of this module, students are expected to be able to:

- 1 Demonstrate an understanding of core programming techniques and concepts.
- 2 Select appropriate data structures and coding techniques to solve a simple problem.
- 3 Identify and use existing libraries relevant for solving a given problem.
- 4 Apply industry standard version control tools to manage software deliverables.
- 5 Describe and explain the software development process as aligned to industry practice.

Indicative Module Content

Introduction to Programming Logic and Design: The programming environment; object-oriented programming concepts; Variables and Data Types; control structures (conditionals, loops); working with lists and strings; input and output; functions; classes; files, interpreting errors and exceptions; testing and simple debugging of code. Object oriented programming. Reusing existing functionality: libraries and APIs: developing and/ or extending existing solutions within a business analytics context. Tools and techniques for sharing software solutions. Version control. Software development life cycle.

Module Delivery

The module is delivered in Blended Learning mode using structured online learning materials/activities and directed study, facilitated by regular online tutor support. Workplace Mentor support and work-based learning activities will allow students to contextualise this learning to their own workplace. Face-to-face engagement occurs through annual induction sessions, employer work-site visits, and modular on-campus workshops.

Indicative Student Workload	Full Time	Part Time
Contact Hours	30	N/A
Non-Contact Hours	30	N/A
Placement/Work-Based Learning Experience [Notional] Hours	240	N/A
TOTAL	300	N/A
<i>Actual Placement hours for professional, statutory or regulatory body</i>	240	

ASSESSMENT PLAN

If a major/minor model is used and box is ticked, % weightings below are indicative only.

Component 1

Type:	Coursework	Weighting:	100%	Outcomes Assessed:	1, 2, 3, 4, 5
Description:	This coursework will consist of a software design and development exercise and a presentation on aspects of software development lifecycle as demonstrated within the workplace environment.				

MODULE PERFORMANCE DESCRIPTOR

Explanatory Text

The module is assessed on a pass/unsuccessful basis. The Module Grade is based on performance in Component 1 (coursework) and Component 2 (exam) as detailed below.

Module Grade	Minimum Requirements to achieve Module Grade:
Pass	Pass in Component 1
Fail	Fail, i.e. unsuccessful, in Component 1
NS	Non-submission of work by published deadline or non-attendance for examination

Module Requirements

Prerequisites for Module	None, in addition to course entry requirements.
Corequisites for module	None.
Precluded Modules	None.

INDICATIVE BIBLIOGRAPHY

- 1 LUTZ, M., 2013. Learning Python. Beijing, China: O'Reilly.
- 2 PADMANABHAN, T.R., 2016. Programming with Python. Singapore, Singapore: Springer.
- 3 HETLAND, M.L., 2017. Beginning Python: from novice to professional. 3rd ed. New York, NY: Apress.
- 4 SOMMERVILLE, I., 2016. Software engineering. 10th ed. Harlow: Pearson.