H06OOP: Object Oriented Programming

Module Code:)P			
Long Title		Object Oriented Programming APPROVED			
Title		Object Oriented Programming			
Module Level:		LEVEL 6			
EQF Level:		5			
EHEA Level:		Short Cycle			
Credits:		5			
Module Coordinator:		FRANCES SHERIDAN			
Module Author:		ANCES SHERIDAN			
Departments:		nool of Computing			
Specifications of the qualifications and experience required of staff		begree in Computing or cognate discipline, or the equivalent experience in industry as programmer.			
Learning Outcomes					
Learning Outco	mes				
On successful co	mes ompletion of this modu	le the learner will be able to:			
On successful co	mes ompletion of this modu Learning Outcome	lle the learner will be able to: Description			
On successful co	mes ompletion of this modu Learning Outcome Apply theoretical cor	lle the learner will be able to: Description Icepts to a range of contexts and problem domains			
On successful co # LO1 LO2	Description of this module Learning Outcome Apply theoretical cor Formulate computer	le the learner will be able to: Description neepts to a range of contexts and problem domains program solutions to well defined abstract problems			
Ceanning Outco On successful co # LO1 LO2 LO3	Apply theoretical con Formulate computer Design, develop, tes	le the learner will be able to: Description ncepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs			
Consuccessful co # LO1 LO2 LO3 LO4	Apply theoretical composition of this modules of the second secon	Ile the learner will be able to: Description neepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			
Cearning Outco On successful co # LO1 LO2 LO3 LO4 Dependencies	Apply theoretical completion of this module Learning Outcome Apply theoretical computer Formulate computer Design, develop, tes Identify and discuss	Ile the learner will be able to: Description Incepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			
Cearning Outco On successful co # LO1 LO2 LO3 LO4 Dependencies Module Recomm	Apply theoretical completion of this module Apply theoretical computer Formulate computer Design, develop, tes Identify and discuss mendations	Ile the learner will be able to: Description Incepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			
Consuccessful co mathematical Consuccessful co mathematical LO1 LO2 LO3 LO4 Dependencies Module Recommenda	Apply theoretical completion of this module Apply theoretical computer Formulate computer Design, develop, tes Identify and discuss mendations tions listed	Ile the learner will be able to: Description Incepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			
Co-requisite Module	Apply theoretical completion of this modules	Ile the learner will be able to: Description Incepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			
Co-requisite McCo-requisite McCo-req	Apply theoretical completion of this modules listed	le the learner will be able to: Description neepts to a range of contexts and problem domains program solutions to well defined abstract problems t and debug moderately complex object-oriented programs object-oriented design principles and design patterns in a given object-oriented design.			

H06OOP: Object Oriented Programming

Module Content & Assessment					
Indicative Content					
Inheritance Introduction to Inheritance • The role of reuse and inheritance					
Polymorphism • How to utilize polymorphic constructs in programming					
Polymorphism and Inheritance • Use of support libraries from external sources • Revision of Polymorphism and Inheritance					
Array Lists • Arrays of Objects • Introduction to Java Co	ollection Framework Using ArrayLi	ists in Java • Two-Dimensional Arrays			
File Input and Output File Input and Output File and FileDialog Object. • Low-Level File I/O. • High-Level File I/O. • Creating and Handling Exceptions. • Object I/O.					
Principles of good Object Oriented Desig • Object Programming Principles • Object O	Principles of good Object Oriented Design Object Oriented Design Object Oriented Design				
Design Patterns (e.g. Decorator Pattern) • Behavioural Design Patterns (e.g. Singleton Pattern) • Structural Design Patterns (e.g. Singleton Pattern) • Structural Design Patterns (e.g. Decorator Pattern) • Behavioural Design Patterns (e.g. Decorator Pattern)					
GUI Objects and Event Driven Programm • GUI Objects • Positioning GUI Objects • E	ning Event Handling • Handling Multiple	Events			
Regular Expressions • Introduction to Regular Expressions • Dev regular expressions	reloping programs for data process	sing activities (e.g., data extraction, cleaning, m	erging, aggregation, analysis, reporting) using		
Software Testing • The importance of testing • Methods of test	sting • Writing a Unit Test • Precor	ditions and post conditions • Black Box and Wh	ite Box		
OOP Programming other than Java Java vs Programming Language					
File I/O & Exception Handling • File and FileDialog Object • Low-Level File	e I/O • High-Level File I/O • Creatir	ng and Handling Exceptions • Object I/O			
Data Connectivity • Database Programming - CRUD					
OOP Programming in another programming language • Java vs Another Programming language					
Assessment Breakdown			%		
Coursework			100.00%		
Assessments					
Full Time					
Coursework					
Assessment Type:	Continuous Assessment	% of total:	40		
Assessment Date:	n/a	Outcome addressed:	1.2.3.4		
Non-Marked:	No		, , , ,		
Assessment Description:					
Each week student will submit program code to the Moodle server for grading. Student will be supplied with an interface specification for the program(s) and the grading will be conducted via automated unit testing based on unknown inputs. Students will be examined on their ability to convey understanding of the programs which they have developed.					
Assessment Type:	Project	% of total:	10		
Assessment Date:	n/a	Outcome addressed:	1,2,3,4		
Non-Marked:	No				
Assessment Description: Students will work in groups to create an OOP programming in another language other than Java.					
Assessment Type:	essment Type: Practical % of total: 50		50		
Assessment Date:	n/a	Outcome addressed:	1,2,3,4		
Non-Marked:	No				
Assessment Description: The students will have to develop solutions to programming problems relevant to all material covered in the module using a proctored computer in an examination environment. There will be a written component to assess the student ability to determine errors in a program.					
No End of Module Assessment					

No Workplace Assessment

Reassessment Requirement

Coursework Only This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination.

Reassessment Description
This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination. Project that meets all learning outcomes is provided to the student.

H06OOP: Object Oriented Programming

Module Workload							
Module Target Workload Hours 0 Hours Workload: Full Time							
Lecture	No Description		24	Every Week	24.00		
Lab	No Description		24	Every Week	24.00		
Independent Learning	No Description		77	Every Week	77.00		
Total Weekly Contact Hours					48.00		
Workload: Part Time							
Workload Type	Workload Description		Hours	Frequency	Average Weekly Learner Workload		
Lecture	No Description		24	Every Week	24.00		
Lab	No Description		12	Every Week	12.00		
Independent Learning	No Description		89	Every Week	89.00		
Total Weekly Contact Hours				36.00			

Module Resources				
Recommended Book Resources				
Vaskaran Sarcar. (2016), Interactive Object Oriented Programming in Java, Apress, p.211, [ISBN: 978-1-4842-2543-1].				
Supplementary Book Resources				
Herbert Schildt. (2017), Java: The Complete Reference, Tenth Edition, McGraw-Hill Education, p.1344, [ISBN: 978-1259589331].				
This module does not have any article/paper resources				
This module does not have any other resources				
Discussion Note:				