MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Python Language Programing			**Module Delivery**	
Module Type	**Basic**			☒ **Theory** ☐ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	ATU11612				
ECTS Credits	**3**				
SWL (hr/sem)	**75**				
Module Level	UGIII		**Semester of Delivery**	2	
Administering Department			**College**	NETC	
Module Leader			**e-mail**		
Module Leader's Acad. Title			**Module Leader's Qualification**		
Module Tutor	Name (if available)		**e-mail**		
Peer Reviewer Name	Nasir Hussein Selman		**e-mail**	Coj.nas@atu.edu.iq	
Scientific Committee Approval Date	01/05/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	1. To define the fundamental concepts of Python programming. 2. To introduce the principles of coding 3. To learn problem-solving using Python. 4. To learn various programming techniques and structures in Python.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Understand the fundamental concepts, syntax of Python, write and execute scripts Python. 2. Utilize variables, data types & operators, loops and conditional statements. 3. Define and use functions to create modular code. 4. Work with data structures like lists, tuples, dictionaries, and sets.

	5. Handle files and perform basic input/output operations.
Indicative Contents المحتويات الإرشادية	1. Syntax of Python 2. Loops and conditional statements 3. Data structures 4. Input / Output operations.

Learning and Teaching Strategies
استراتيجيات التعلم والتعليم

Strategies	The main strategy that will be adopted in delivering this module is to encourage students' participation in coding exercises, while at the same time refining and expanding their problem-solving and critical thinking skills. This will be achieved through interactive lectures, hands-on coding tutorials, and engaging programming tasks that involve real-world applications and challenges relevant to the students.

Student Workload (SWL)
الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	63	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	12	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**75**		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	LO #1, #2 and #10, #11
	Assignments	2	10% (10)	2 and 12	LO #3, #4 and #6, #7
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	LO #5, #8 and #10
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #7
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Introduction to Python and Computer Programming; Python as a programming tool; Different Python versions; Setting up the Python environment
Week 2	Data Types and Variables; Writing your first Python program; Understanding literals and variables; Basic input and output operations
Week 3	Basic Operators; Arithmetic, comparison, and logical operators; Operator precedence and associativity
Week 4	Conditional Execution; Implementing conditional statements; Nested conditions and conditional expressions
Week 5	Loops; Understanding and using 'while' and 'for' loops; Loop control mechanisms: break, continue, and else clauses
Week 6	Lists and List Processing; Creating and modifying lists; List operations and methods; Sorting algorithms, including bubble sort
Week 7	Logical and Bitwise Operations; Boolean logic in Python; Bitwise operators and their applications
Week 8	Functions; Defining and invoking functions; Function parameters, arguments, and return values; Scope and lifetime of variables
Week 9	Tuples and Dictionaries; Creating and using tuples; Understanding dictionaries and their methods
Week 10	**Mid-Term Exam**
Week 11	Exception Handling; Understanding errors and exceptions; Try-except blocks; Handling multiple exceptions and using 'finally'
Week 12	Modules and Packages; Importing and using modules; Exploring Python packages; Standard libraries and third-party modules
Week 13	String and List Methods; String operations and methods; Advanced list methods and comprehensions
Week 14	Course Review; Comprehensive review of all topics; Hands-on project to apply learned concepts; Preparation for PCEP and PCAP certifications
Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Installing Python, Exploring Python versions, Writing and running a basic Python program
Week 2	Declaring and using different data types (int, float, string), Understanding literals and variables, Implementing basic input and output operations
Week 3	Writing programs that use arithmetic, comparison, and logical operators, Understanding operator precedence and associativity
Week 4	Implementing if, elif, and else statements, Writing programs with nested conditions, Using conditional expressions to simplify code
Week 5	Creating programs with 'while' and 'for' loops, Implementing loop control mechanisms like break, continue, and else clauses
Week 6	Writing programs that create and modify lists, Using list operations and methods, Implementing sorting algorithms like bubble sort
Week 7	Writing programs that use Boolean logic, Implementing bitwise operators for efficient data manipulation
Week 8	Defining and invoking functions, Passing arguments and returning values, Understanding function scope and lifetime of variables
Week 9	Writing programs to create and use tuples, Implementing dictionaries and utilizing their methods for efficient data storage and retrieval
Week 10	**Mid-Term Exam**
Week 11	Writing programs that use try-except blocks, Handling multiple exceptions, Using 'finally' for clean-up operations
Week 12	Creating and importing custom modules, Exploring Python packages, Using standard and third-party libraries in programs

Week 13	Implementing string manipulation and advanced list operations, Writing programs using string methods, List comprehensions for efficient data processing
Week 14	Developing a comprehensive project that incorporates all course topics, Applying learned concepts to build a functional Python application, Preparing for PCEP and PCAP certification exam scenarios
Week 15	Course Review
Week 16	**Final exam**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	**"Python Crash Course" by Eric Matthes**	yes
Recommended Texts	**"Automate the Boring Stuff with Python" by Al Sweigart**	no
Websites	https://realpython.com/	

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

Note: Marks Decimal places above or below 0.5 will be rounded to the higher or lower full mark (for example a mark of 54.5 will be rounded to 55, whereas a mark of 54.4 will be rounded to 54. The University has a policy NOT to condone "near-pass fails" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.